REPORT DOCUMENTATION PAGE

AFRL-SR-AR-TR-04-

0525

1. REPORT DATE (DD-MM-YYYY)	2. REPORT TYPE	3. DATES COVERED (From - To)
09/20/2004	Final Technical Report	02/15/2002-04/30/2004

4. TITLE AND SUBTITLE
A High Resolution Aeroelasticity Method for Fighter Aircraft

At Flight Reynolds Numbers.

5a. CONTRACT NUMBER

5b. GRANT NUMBER
F49620-02-1-0117

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Squires, Kyle D.

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Mechanical and Aerospace
Engineering Department,
Arizona State University,
Tempe, AZ 85287-6106

8. PERFORMING ORGANIZATION REPORT NUMBER

XAA0126/TE

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Air Force Office of   ATTN: Thomas Beutner
Scientific Research
4015 Wilson Blvd
Rm 713
Arlington, VA 22203-1954

10. SPONSOR/MONITOR'S ACRONYM(S)
AFOSR/NA

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

APPROVED FOR PUBLIC RELEASE.

13. SUPPLEMENTARY NOTES

20041101 143

14. ABSTRACT

The principle objective of the present study was the application and assessment of Detached-Eddy Simulation for predicting the massively separated flow around forebodies. Computations were performed of the flow around both stationary and rotating forebodies at a Reynolds number of 2.1 million and at angles of attack of 60 and 90 degrees. While RANS predictions at 60 degrees angle of attack were adequate for the mean pressure distribution, at 90 degrees angle-of-attack, RANS yielded coherent vortical structures along the forebody that result in substantially greater variations in the surface pressure than measured. DES predictions accurately accounted for the chaotic structure in the wake; predicted pressures were in good agreement with measurements. For the forebody undergoing prescribed rotary motion at a spin coefficient of 0.2, DES predictions were mostly adequate though with some discrepancies between predicted and measured pressures. Possible causes for the discrepancies and discussion of other issues important to successful applications of DES are also summarized in this report. In the supplement, RANS and DES were applied to prediction of the separated flow over a surface-mounted hump, the objective being an assessment of the techniques in an application relevant to flow control. Detailed comparisons of the RANS and DES against experiments for three configurations are reported.

15. SUBJECT TERMS
turbulence simulation and modeling, separated flow, Detached-Eddy Simulation

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE		73+48	Kyle D. Squires
					19b. TELEPHONE NUMBER (include area code) (480) 965-3957

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std. Z39.18

A High Resolution Aeroelasticity Method for Fighter Aircraft at Flight Reynolds Numbers

Kyle D. Squires
Mechanical and Aerospace Engineering Department
Arizona State University
P.O. Box 876106
Tempe, AZ 85287 USA

ASU ARIZONA STATE UNIVERSITY

Executive Summary

Accurate prediction of the flowfields around fighter aircrafts at high angles-of-attack is of great engineering interest. For modern fighters, the flow around the forebody is complex and significantly contributes to the overall forces and moments experienced by the airplane. Accurate predictive techniques are important, though current engineering approaches based on solution of the Reynolds-averaged Navier-Stokes (RANS) equations appear deficient and unable to represent the complex physics of massively separated flows at high Reynolds numbers to sufficient accuracy. Large-Eddy Simulation (LES) provides a more realistic treatment of the separated regions of a turbulent flow but is prohibitively expensive when applied to whole domains at high Reynolds numbers. Detached-Eddy Simulation (DES) is a hybrid method, combining RANS and LES, and attempts to take advantage of both techniques in regions where each is accurate and computationally feasible. In natural applications of the method, attached boundary layers are entrusted to the RANS model with detached regions of the flow predicted using LES. The principle objective of the present study was the application and assessment of DES for predicting the massively separated flow around forebodies.

Computations were performed of the flow around both stationary and rotating forebodies at a Reynolds number of 2.1×10^6, based on the freestream velocity and body width (diameter). Most of the computations were performed at an angle-of-attack of 90°, with some simulations also performed at a 60° angle-of-attack. The calculations were performed on unstructured grids using *Cobalt*, a compressible-flow Navier-Stokes solver. For the flows at a 60° angle-of-attack, RANS predictions were in good agreement with the measured pressure distributions at eight axial stations along the forebody. At 90° angle-of-attack, owing to the large amount of flow separation, RANS predictions were not accurate, yielding coherent vortical structures along the forebody that result in substantially greater variations in the pressure than measured. DES predictions accurately accounted for the chaotic structure in the wake; predicted pressures were in good agreement with measurements. For the forebody undergoing prescribed rotary motion at a spin coefficient of 0.2, DES predictions were mostly adequate though with some discrepancies between predicted and measured pressures. Possible causes for the discrepancies and discussion of other issues important to successful applications of DES are also summarized in this report.

Contents

1 Introduction

1.1 Motivation

Many Air Force aircraft operate in regimes characterized by high angle of attack and high Reynolds numbers. The flow is unsteady and maneuvering in these conditions is characterized by rotation, non-linearity, and unsteadiness. The forebody, owing to its long moment arm, is an important contributor to spin characteristics and a comprehensive knowledge of the flowfields encountered is required for design of strategies to minimized and control aspects such as aircraft spin.

Flowfield analysis has traditionally been carried out using wind-tunnel and flight tests. The flowfields are often studied using smoke visualizations, while the surface flow is studied using dye or oil applied to surfaces during flight. With rapid advances in hardware and software, computational analysis has gained popularity owing to the ease with which several configurations can be tested. Though promising, simulation tools continue to require enhancements, especially in the predictions of flows experiencing massive separation.

Most engineering predictions are obtained from Reynolds-averaged Navier-Stokes (RANS) approaches. RANS approaches are economical though subject to substantial empirical input. The approach is generally realized today to be reliable in thin shear layers, where the turbulence models have been calibrated, but the predictions are inconsistent for flows characterized by massive separation. Though unsteady RANS (URANS) models have become a topic of increasing interest, such approaches have not been demonstrated to provide a consistent increase in accuracy when applied to massive separations.

The relatively poor performance of RANS models motivated the increased application of Large-Eddy Simulation (LES). LES is a powerful approach to directly representing the large eddies which are dependent on the geometry and the boundary conditions. LES filters the small scales of motion and models their effect on the resolved scales. The computational cost of the technique is essentially independent of the Reynolds number in regions away from the solid surfaces. However, the need for very fine grids in boundary layers raises the cost, so much so that applications at flight conditions are decades away (Spalart 2000).

Detached-Eddy Simulations (DES), is a hybrid method proposed by Spalart (1997) which entrusts the region near the walls to RANS models and switches to LES away from the wall, taking advantage of the efficiency of RANS in the boundary layer and the resolution of LES in separated regions. The formulation is obtained by a simple alteration of the length scale of the destruction term in the Spalart-Allmaras (S-A) RANS equation Spalart-Allmaras (1997). This alteration increases the destruction term in the S-A RANS equation, drawing down the eddy viscosity. The reduction in the eddy viscosity allows instabilities to develop in this region, like in a classical LES approach. This technique is not only feasible for high Reynolds number flows but also resolves three dimensional, time dependent turbulent motions. The region of transformation from RANS

to LES is fixed by the grid and so the technique is non-zonal. Grids can be generated to entrust the whole or a major part of the boundary layer to RANS. The LES character of the method is achieved by effectively introducing the grid spacing into the model. The natural applications of the technique are in massively separated flows, similar to the case under consideration, with a fully turbulent boundary layer. Turbulence dynamics in the boundary layer that are damped by the RANS treatment are not an important error source in this class of flows.

The principle objective of this study is to assess URANS and DES predictions of the flowfields around forebodies. The flowfields are characterized by turbulent separation and owing to the smoothness of the geometry, separation prediction is a non-trivial challenge posed to the modeling. A case with rotary motion has been studied which imposes additional complexity.

1.2 Related experimental and computational studies

The flow around full aircraft have been studied using both experimental techniques (Fisher *et al.* (1985), Bjarke *et al.* (1985)) as well as computationally (Forsythe *et al.* (2002)). Experiments by Bjarke *et al.* (1985) and computational predictions by Forsythe *et al.* (2002) showed the contribution of the forebody to the moments and forces acting on the aircraft. Computational analysis of a two-dimensional forebody cross-section was carried out by Squires *et al.* (2001), for laminar and turbulent boundary conditions. This analysis performed using URANS, DES and LES showed the robustness and accuracy of the DES in predicting the turbulent cases. Additionally, the utility of unstructured grids in optimizing grid generation for turbulence-resolving computations was also noted.

Rotary balance experiments on square and circular ogive forebodies were reported by Pauley *et al.* (1995), which established an extensive database for various Reynolds numbers and rotation rates. The forebodies were tested at high angles of attack, $\alpha = 60°$ and $90°$ and Reynolds numbers ranging from 8×10^4 to 2.25×10^6. For cases with rotary motion, the spin coefficients ($\Omega L/(2U_\infty)$) were varied between ± 0.4. Measurements were acquired for the pressure distribution around the bodies at eight axial stations along the body. Also reported were the yawing moment and the side forces coefficients.

Pauley *et al.* (1995) found a strong correlation of the flow attachment with the local Reynolds number. Since this varied with the width of the forebody the measurements showed that a Reynolds number of at least 2×10^5 was required so that the flow remains attached on the forebody. Also they observed that the yawing moment and spin coefficient did not exhibit the same characteristics. In addition to the yawing moments and the spin coefficients, pressure coefficients were also reported.

van Dam *et al.* (2001) computed one of the cases measured by Pauley *et al.* (1995) at $60°$ angle of attack and a spin coefficient of 0.2. Most of the calculations were done using a URANS approach and with the Baldwin-Lomax model. Some additional calculations were carried out using Spalart-Allmaras model,with good agreement reported between the calculations and the experiments. The

2

simulations helped to exhibit the feasibility of RANS in such flows.

1.3 Approach

The flow around a rectangular ogive forebody cross-section is considered (Figure 1-2). The length of the forebody is twice the width D. The aftbody extends further to a length of $4D$. The cross-section is a square with rounded corners, with the corner radius being $1/4$ of the width, resembling the cross-section of the X-29 and T-38. The rounded corners of the forebody pose a challenge to the model in predicting the point of separation, unlike cases with sharp corners in which the geometry fixes separation.

Unstructured grids find applicability in cases involving complex geometries as the time alloted to grid generation can be reduced. Additionally, it is straightforward to accommodate grid adaption and grid motion using unstructured grids. Unstructured grids were used in previous DES applications by Forsythe (2000).

In the present effort, a structured background mesh (Pirzadeh (1993)) is used to define the grid parameters which yields an optimum distribution and good control of the grids within the domain. Grids are generated using a hybrid advancing layers - advancing front technique (Pirzadeh (1993)). Grid parameters were defined based on the requirements for a DES simulation as defined by Spalart (2001). The cells were compressed in the direction of the velocity gradient in the boundary layer, yielding prism cells, and more isotropic tetrahedral cells in the LES regions away from the wall. An unstructured, implicit flow solver based on the Godunov Riemann method Strang *et al.* (1999) is used to solve the governing equations, using the Spalart-Allmaras RANS and a DES models to model the turbulence. The numerical method is a finite volume cell-centered approach which is second-order accurate in space and time.

Pauley *et al.* (1995) noted that the boundary layer is fully turbulent for the high Re cases considered in their experiments, i.e., for Reynolds greater than about 1.0×10^6. Fully turbulent simulations are produced in the computations by introducing a small level of eddy viscosity at the inlet of the computational domain, sufficient to activate the turbulence model as the fluid enters the boundary layers. The simulations were performed in a cubic domain which is 20 times the length of the body. The surface of the body was defined as an adiabatic no-slip boundary condition while the outer domain was defined as a farfield boundary.

In the current study, a baseline grid with approximately 6.5×10^6 cells has been used to compute the flow for the various cases: angles of attack of $60°$ and $90°$ for stationary configurations and the case with rotary motion at $90°$ angle of attack. The calculated results were compared with the experimental measurements and the performance of the models analyzed for each of the cases. Simulations were also carried out using a coarse grid with 2.1×10^6 cells and a denser grid comprised of 8.75×10^6 cells in order to determine the sensitivity of the model to grid density. Based on the freestream velocity and the width D of the body the solutions were sampled for 100 time

units using a dimensionless timestep of 0.025.

2 Grid Generation – Procedure

Grids are broadly classified as either structured or unstructured. Structured grids involve a regular lattice structure arising from discretizing the domain. The whole lattice is then transformed into a computational domain in a Cartesian form. However, gridding domains enclosing complex geometries using structured grids can prove to be a relatively inefficient and time-consuming task. These and other considerations motivate the use of unstructured grids, and the development and application of unstructured meshes comprises a key aspect of the present effort. The basic data structure of an unstructured grid differs from that corresponding to a structured grid, one outcome being that it is in fact less simple to transform an unstructured grid into a computationally simple domain. However, unstructured grids offer other advantages as summarized by Strang *et al.* (1999). These advantages include that the time for grid generation around complex geometries can be reduced and that unstructured methods offer a relatively simple path for grid adaption and grid motion compared to structured grids. Unstructured grids also suffer from some disadvantages owing to the irregular data structure. These disadvantages include the fact that flow solvers for an unstructured grids are usually more expensive, i.e., carry higher computational cost, than structured solvers and that grid quality (e.g., cell skewness) may not be comparable to that on structured grids.

Owing to the reduction in grid generation time and solution time (because of parallel processing in the present effort) and flexibility to adaption and grid motion an unstructured grid has been preferred over a structured grid for the present effort. Grids have been generated in VGRIDns, an unstructured grid generation software based on advancing layers - advancing front technique, and the flow solved using *Cobalt*.

Grid generation, in VGRIDns, essentially consists of the following steps:

1. Creating the basic geometry (IGES file).

2. Dividing the geometry into patches (three dimensional polygons) using GridTool.

3. Projecting the patches onto a surface (i.e., associating patches with a surface).

4. Defining the spacing of the grids by defining the sources used in GridTool.

5. Generating the surface patches using VGRIDns.

6. Projecting the surface patches using Projector program.

7. Saving the updated rst, d3m, front files.

4

8. Generating the volume grid using VGRIDns.

9. Post processing the grid using Postgrid.

More details on the above procedures are summarized below.

2.1 Loading the geometry

The basic geometry is created as a CAD file in a CAD package such as Solidworks and is exported in a GridTool-readable format (IGES, P3D or GridGen). For simpler geometries, a code can also be written to create a file. The file is loaded in GridTool in the same format. Any modification to the geometry should be accomplished prior to loading the file in GridTool. The surfaces imported are checked for errors. The surfaces loaded can be checked using the Surface panel available in GridTool.

2.2 Creating curves

After loading the geometry, curves must be created on the surfaces. Before the curves are laid out it has to be ensured that the created curves lie on the surface. Thus, the "On Surface" button was active when the curves were created. The easiest option to create curves is to use the "Auto Edge" button in the "Points and Curves" menu. This lays out the curves on the surface of the skeleton and thus saves significant processing time.

The curves can also be created manually by positioning the end points of the curve on the surface of the geometry. This process is, however, cumbersome. This is done by using the "Next Curve" button to initialize the curve. The curve is assigned a number which is used to subsequently refer to it. The "Next Point" button is then used to initialize the starting point of the curve and the point selected. Selection of the point can be done by placing the cursor on the required point and using the hot key "p". Alternatively it can be positioned using the "U&V Panel" in the lower right corner of the menu. Thus, once the curve has been started, it can be completed by either proceeding point by point or by defining the end point and allowing the software to determine the intermediate points on the curve. To proceed point by point the "Next Point" button was chosen again and the next point defined. This is the most effective way to build the curve in case of complex geometries. In case of simpler geometries, the last point can be defined and the "Enrich" button can be used to allow GridTool to determine the intermediate points automatically. This saves effort in building the curve. It should be ensured that the curve passes through the desired path while using the Enrich option. In case of any problems, the part of the curve can be formed by individually selecting the points. Also, a combination of the two can be used while constructing long curves, i.e., by breaking the longer curve into smaller curves to define the points and using the Enrich option. Care must

be taken to project the points of the curve using "Project Curve" before the Enrich command was used, in order that VGRIDns creates all the points on the surface.

The curves formed are represented in green. A selected curve is displayed in red with the points in red or green. The selected point on the curve is red while all the other points are green. Once the curves are created it is recommended to check for the presence of repeated curves. This can be done by going through all the curves by browsing from the first to the last curve using the curve numbers in the "Points and Curves" menu if the geometry is not complicated. Alternatively this can also be done by clicking on each of the curves using the hot key "c" for selecting the curves and checking the curve number that appears in the number box. Different numbers appear in the box if curves are superimposed. The unnecessary curves are deleted. Checking for the presence of repeated curves at this stage ensures the grid will be grown on a smooth surface.

2.3 Creating patches

VGRIDns recognizes three types of patches:

- Triangular Barnhill - Gregory-Nielson Patches (Three arbitrary sides)

- Bi-linear transfinite Coons patch (Four arbitrary sides)

- Planar patch (n sides all lying in a single plane)

Patches are enclosed surfaces formed from curves. The "Patch" menu in GridTool is used to create/delete/edit patches. The "Auto Patch" command is the simplest way to create patches. For each of the surfaces VGRID generates four patches by dividing the surface about the center. When "Auto Patch" was used the surfaces were automatically assigned to the patches.

If further subdivisions of a surface were required the patches were created manually. To start a new patch, the "Next Patch" option was chosen from the menu. This initialized the patch and a patch number was assigned to it. The first curve of the patch was selected using the hot key "c" for the curve or by selecting the curve using the curve number. The curve was added to the patch using the "Accept Edge" option. The next curve in the patch was selected using the "Find Edge" option or by selecting the curve. The patch is closed by continuously accepting the required curves. The information in the lower right corner of the "Patch" panel is helpful in determining if the patch is closed or not. All the patches must be closed for the creation of an acceptable d3m file, which is the file used to generate the grids. After all the curves are patched, a cuboidal domain was generated around the geometry using the "Box" option. The maximum and minimum x, y, z coordinates of the box are input in the respective fields to create the box.

Incomplete patches resulting from the above process are represented with blue dotted lines. The arrow indicating the direction in which the grid will grow is shown in pink. As the patch is

6

completed it is represented by solid blue lines and a solid pink arrow. The direction of the arrow by default is normal to the patch.

The "right hand rule" is useful handy in creating patches (3). If the fingers of the right hand are curled in the direction (clockwise or counter clockwise) in which the patch is being formed, the thumb points in the direction in which the grid grows. The arrow is usually normal to the patch. The orientation can however be altered, if necessary, using the "Rotate Patch" option in the menu. The grid should grow outwards in case of a solid body and inwards in case of the outer box. The direction of the patch can also be reversed in one step using the "Reverse Patch" option. Also for patches with multiple loops, the outer loop controls the direction of the grid growth. The inner loops should be oriented in the direction opposite to the outer loop.

The patches are assigned a "Family Name". The set of patches with the same set of boundary conditions were designated as the same family. By default this name is "Addams". The name can be changed as necessary by keying in the name and applying the family using "BC / Apply Family" button. If a name is present in the "Patch Family" input field while creating a patch, then the name is automatically assigned to the patch. The family names are helpful in later stages while assigning boundary conditions to the patches in the flow solver.

After the patches are created the corresponding surfaces are associated to them. The patch and the corresponding surface are switched on and the "Accept Surface" command was used to associate the surface to the patch. The associated surface and the patch are saved in the the the .mapbc file. Also, the number of surfaces associated is displayed in the bottom right of the Patch command box. A patch which has been improperly associated can be corrected by re-associating the correct surfaces.

Before proceeding, the bad patches should be checked using the bad patch option. By clicking this option the button turns green and only the bad patches appear on the Display Screen. In general the message in the status box can be used to determine the problems with the patches. The following are the numbers in the status box representing possible problems:

1. Curve repeated

2. Patch not closed

3. Patch is n-sided and not planar

4. No surface associated to the patch

5. Insufficient number of points in the patch

2.4 Placing sources

Grid growth in VGRIDns is based on structured background source allocation Pirzadeh (1993), by which the size of the grids in a region is determined using an elliptic relation, influenced by

7

all the sources in the vicinity. Size of the sources (s), the intensities (an) and the directional bias (bn) influence the grid sizes in the region (Figure 4-5). An efficient way to achieve a desired concentration of grids for a desired configuration as suggested by Pirzadeh (1993) is to experiment with these terms.

A structured background grid distribution ensures that as the grid progresses in space, the grid parameters defining the position of the new grid point are interpolated from the values defined at the point where the source is placed. This type of unstructured grid generation, unlike the traditional methods of unstructured grid generation, allows an elliptical variation in the grid sizes. The advantages of such a scheme include,

- The background grid can be easily generated using minimal user time and effort.

- A smooth, controllable distribution is obtained.

- Modification of the grid is convenient.

A Cartesian background grid is assumed (Figure 4). Two types of sources – nodal and linear – are used. The sources can be placed anywhere in the field, but are usually placed either near the surface or near the outer domain boundaries. These source elements propagate the spacing parameters systematically within the domain.

The spatial variation in the domain is similar to the heat diffusion from discrete sources. The process can be modelled by a Poisson equation, with Dirichlet boundary condition.

$$\nabla^2 S = G, \qquad S = S_b \qquad (1)$$

where S_b is the prescribed spacings, S is the grid spacing parameter and G is the source term maintaining the grid spacings.

The elliptic partial equation is solved on the Cartesian grid to determine the size of the grids, in the vicinity. The concentration of grid points in a region is dependent on the source sizes (s) and intensities (strengths - an,bn).

The source term on each Cartesian grid point is given by

$$G_{ij} = \sum_{n=1}^{N} \psi_n f(r_n, l_n), \qquad (2)$$

where (i, j) represents the background node index, N is the total number of sources, ψ is the intensity parameter, r_n is the distance of the node from the source, and l_n is the length of the source (for nodal sources $f = f(r_n)$). A Gauss-Seidal iterative scheme is used to determine the grid spacing parameters.

8

The parameter s determines the source size, while an and bn denote the intensities, an provides an equal grid distribution around the grid while bn introduces a directional bias in the grid as shown in Figure 5.

The "Next Source" option is used to start a new source. As the "Background Grids" menu opens the "Points & Curves" menu also automatically opens. This helps in placing the sources in any desired location in the three-dimensional space as the co-ordinates of the source can be directly typed in the X, Y, Z co-ordinate boxes. The sources are placed in the desired locations. The family name of the source are assigned. The sources were placed in such a way that the whole surface of the body in consideration is influenced by the sources. This is accomplished by adjusting the size "s" and strengths an and bn of the sources. Similarly the sources are put at the vertices of the computational domain, i.e., the outer boundaries of the domain.

The size of the source "s" determines the number of cells that VGRIDns will generate. Larger source sizes lead to smaller number of cells as the volume is occupied by larger grids and vice versa. In case of stretched sources "S" denotes the stretched length. The size of the sources were adjusted depending on the size of the body and on the desired number of cells.

The value of an is automatically assigned by VGRIDns if it is left as "1" in GridTool. The values an and bn are the intensity parameters. The larger are these values, the greater is area that they will influence. These values are scaled based on the sizes of the sources, i.e., the larger the sources, the higher the values of an and bn. The length of the linear source also determines these values.

To ensure the creation of viscous layers the "Viscous" field in the "Global" sub-menu is changed to 1. For an inviscid grid this could be maintained at 0. No stretching is applied in inviscid grids and therefore the "Stretching" field may be left as 0. These could also be modified later in the d3m file. The first wall normal distance (δ) (e.g., corresponding to $y^+ = 1$) is set using the parameter Delta in the panel. The growth rate is set using Rate1 and Rate2 in the panel.

The distance δ from the wall to the first cell center is determined using the formulae Cobalt Manual (1999),

$$\delta = L\frac{1.3016y^+}{Re_L^{0.75}}, \quad \text{laminar flow} \tag{3}$$

$$\delta = L\frac{(13.1463y^+)^{0.875}}{Re_L^{0.90}}, \quad \text{turbulent flow} \tag{4}$$

At this stage of the mesh-generation process, the .rst, .d3m, .igs and .mapbc files are created. The restart and d3m files are saved from the "Input / Output" menu whereas the iges file and the mapbc file were automatically created when the d3m file was saved. The mapbc file contains information about the various surfaces alloted to each of the patches, while the d3m file contains information about the patches (e.g., boundary conditions, curves forming the patch, etc.) along with information about the curves, points and the sources. The igs file has the basic geometry on which all the curves have been generated. The d3m file can only be saved if there are no errors in

9

the generated geometry. Error or warning messages for bad patches and/or absence of Background Sources flash when the d3m file is saved and these errors must be addressed before the surface grid can be generated in VGRIDns. Also, the different modules of the same project should be combined before the surface grid for the entire configuration is generated.

2.5 Modifying the d3m file

This is an optional step which involves the modification of the d3m file. This is usually used to modify the density of the grid in the whole domain. Modifications in the file can be input by opening the file in a text editor.

The d3m file contains information about the various grid parameters. It also lists the various points on the geometry, the details of curves including the points lying on the curves, the details about the patches with the lines forming the patches and the background grids or sources. The grid parameters can be varied as necessity. The "ifact" and the "vfact" values are used to control the grid intensities. The term "ifact" is used to control the density of the inviscid region or Euler region and "vfact" for the viscous layers or the Navier-Stokes region. The value 1 is assigned to the terms once the d3m file is initially created. The user can coarsen the respective portion of the grid by increasing the value greater than one and refine the grid by decreasing the value to something less than 1. For example, if the value is set to 1.2 the cell size is increased by 20 percent. Similarly, setting the value to 0.8 will yield a cell size that decreases by 20 percent. The effect is noted globally over the grid. Note that while the parameter "vfact" was intended to coarsen or refine the grid it does not have any effect on the mesh. However a change in "ifact" influences the prism layer grids as well as the surface grids generated.

2.6 Generation of surface grids using VGRIDns

Grids in VGRIDns are formed in different steps Garriz (1998). The surface grids must be initially generated and projected properly on the respective surfaces before the volume grid is generated. VGRID can be run in Batch mode or in Interactive mode. The batch mode directly generates the whole mesh without much input from the user, while in the interactive mode the user can view the grids generated and make modifications in the surface grid if required.

The interactive mode is used for generating the grids as patch rotations must be repaired and projection of the surface grid onto the surfaces can be properly ensured. In VGRIDns, the viscous grid type was chosen in order to generate prism layers. As the grid was read from GridTool, VGRID assigns the "an" values to the sources (if not specified by the user). These values were displayed on the UNIX screen, from where VGRIDns is run. Note that VGRIDns crashed at this stage if any errors in the boundary conditions were encountered. The "Background Sources" and the grid point distribution were checked if no errors were encountered.

10

The surface grid (Figure 6) can be generated by gridding patch by patch, displaying each patch in detail to the user. Alternatively, this can also be accomplished automatically and without prompting only if distorted triangulation is encountered. The latter method is more time saving and so was used in this work. Whenever a bad triangulation was encountered, it was displayed as a red triangle with green triangles representing the properly triangulated region. Using the "Esc" button the patch was modified by changing the patch orientation. This usually solved the problem of triangulation. If the problem still existed after all the combinations were tried then the grid parameters such as the source size, strength, etc., were locally modified in order to develop a good surface front. The patch number can be recognized from the UNIX screen which displays the patch triangulation specifications. The user can choose to generate the grid by "Fly" (generation of each grid in the patch) or view the final triangulation. If the number of distorted triangles were small the diagonals were swapped using the hot key 's'. Once all the surface grids had been generated, VGRIDns displays the surface grids. A thorough inspection of the grid was carried out before proceeding further. Any distortions in geometry (however fine it may be), grooves in the grid etc were viewed at this stage and appropriate decisions made to rectify the imperfection. Once inspected, the grid was saved and VGRIDns exit to proceed on to project the grid. The grids formed approximately mimic the geometry but may not exactly lie on the desired surface. So these were projected on the original surfaces. Though the projection step is not necessary for simpler geometries, it is recommended.

2.7 Projection of the surface grid

This step is essential in the grid generation process of complicated geometries. The projection of the surface grid (initial front) generated is carried out to ensure that all the grid points formed lie on the original geometry.

The restart file for the grid is read into GridTool and then the d3m file modified by selecting the d3m (Update) option in the file type input field and reading it. This step is optional and was carried out if changes were made in VGRID while the surface grid was being generated. The changes made by running VGRIDns were updated in the restart file, by writing the new restart file.

To manually project the front the Front (VGRID) file is read in GridTool. Using the Unstructured Grid panel, the front is opened. The patches to be projected are activated using the "Front on/off" switch, the associated surfaces activated using the Surfaces panel, and then using the "Project" switch the front was projected on the surface. The field "dmax" indicates the maximum distance moved by the initial front during projection.

Alternatively the "Projector" program was also used to project the grid. The program was used when projection had to be carried out on flatter surfaces. The program created a new cogsg file and a new front file with extensions cogsgn and frontn respectively. As the projection was carried out the details of the projection were displayed on the UNIX screen. The program automatically stores

the details in the "projector.info" file. The cogsgn file was read into GridTool and was updated using 'Update'. Then the .frontn file was read into GridTool and written as the front file. This updates the front for the algorithm and replaces the existing front with the projected front.

The projection technique is based on a Newton-Raphson method. Most of the geometries are NURBS surfaces, which had been approximated using smaller patches created in GridTool. The points on the curve, obtained from the IGES file (the skeleton), are used as the reference points for the projection. The points generated by VGRID (which are stored in the .front file) are then projected onto the skeleton of the geometry. Abolhassani (1993), suggests five iterations of the Newton-Raphson method, on an average, for a good projection.

At the end of this stage the .bc, .cogsg and the .front files are created.

2.8 Generation of the volume grid

The volume grids in VGRIDns are formed in different layers. The prism grids near solid surfaces approximately cover the boundary layer and are generated by extruding the surface grids normal to the wall. The cells in this region constitute a major fraction of the total number of cells though the size of the domain represented by these cells is typically not large. The viscous grids are extruded from the surface with the initial layer at a distance of "delta" and growing at a rate depending on the values of "Rate1" and "Rate2". Since the prism cells are formed by extrusion these are prism shaped. Immediately following the projection of the initial front the cells are extruded into prisms (Figure 8). Before VGRID proceeds on to generate the viscous grids the "surface vectors" are viewed to check the direction of growth of the prism grids (Figure 11). VGRIDns extrudes the cells until the size of the cells exceed the size determined by the sources defined (Figure 9). VGRIDns is then concluded and the prism layers saved before the inviscid grids are generated.

The inviscid grids cover a larger area of the domain but form the smaller fraction of the total number of cells. The inviscid grids are comprised of tetrahedra. VGRIDns is restarted to generate the inviscid grids. The domain is meshed until VGRID is no longer able to generate cells. The d3m, cogsg, and front files are updated by VGRID after each cell is generated. VGRIDns saves the grid and displays a message indicating the effectiveness with which it has meshed the domain. The three types of messages encountered are listed below:

1. 'Grid is complete. Please post-process the grid for quality improvement.' This indicates that the grid is complete and the complete volume has been filled with cells and no point exists on the current front. The grid as such can then be sent for processing. However the grid is first passed through Postgrid to check for negative or distorted faces.

2. 'No more cells can be formed. Grid may be completed by post-processing.' This is the most frequently encountered case. This indicates that the grid has filled the space to the maximum

possible extent and there exist pockets or vacant spaces which are unfilled. These are usually present in the areas where the grids from various sources meet and are rectified in Postgrid.

3. 'Number of points generated exceeds maximum. Please increase mpoin.' This indicates that the number of cells created have exceeded the maximum limit specified in the d3m file. In such cases the sources and the 'ifact' were modified accordingly so that VGRIDns could handle the size of the grid. Since the limit fixed by VGRID is sufficiently large ($\mathcal{O}(10^7)$), this is less frequently encountered.

After this stage the .bc, .cogsg, and .front files are modified and the .poin1 file created since the viscous layers are generated.

2.9 Post-processing the grid using Postgrid

The steps outlined here are carried out to upgrade the grid quality by filling in gaps and removing any overlapping and skewed cells. Postgrid is used for this purpose. Postgrid is run and the project name entered when prompted. This displays the empty pockets in the domain. Layers near the pockets must be removed by Postgrid before the pocket is filled. Initially in the first iteration the 'Viscous and Inviscid Portions' are removed and then in the subsequent runs layers are removed depending on the region where the void or pockets exist. Once all the pockets are meshed the grid is checked for negative (overlapping) and distorted cells using 'Local Remeshing'. This is also run until the number of distorted cells are reduced to the minimum. The quality is checked in between the iterations to determine the number of distorted cells.

Once a good grid is obtained the grid can be saved and Postgrid exited. At this stage the .cogsg file is modified and updated as the pockets are filled and the bad cells removed.

2.10 Generation of the output and the boundary condition files

The cogsg file is loaded into Blacksmith and the grid converted into a *Cobalt*-readable format. The patch families are assigned patch numbers when prompted. These numbers are used to refer to the families when the boundary condition file is created. The grid is converted into a *Cobalt*-compatible file using the "Grid to Cobalt" option within Blacksmith. No layers in the grid are combined initially and therefore a value of 0 is entered during the conversion when prompted. Once the *Cobalt* file is formed the tetrahedral layers just outside the prism layers are combined using the 'Tet/Pyramid/Prism Layers' command. Thus, the tetrahedral cells in the inviscid region transform smoothly into the prism layers in the boundary layer through an intermediate region of pyramids. This also reduces the total number of cells in the domain and improves the quality of the grid. A 'Soundness Check' is performed to check for the presence of imperfect cells. Any imperfections in the grid are rectified in Postgrid before the cell is sent to *Cobalt* for processing.

13

The *Cobalt*-compatible boundary condition file is generated, after checking for soundness, using the 'Create BC file' option. The families are assigned the appropriate boundary conditions and names. The grid is then passed through the flow solver to obtain the flow solution.

2.11 Processing the solution

The processing of the solution involves defining the various parameters associated with the execution of the flow solver. The data to be specified includes the numerical method, the processor specifications, the system of units, axis systems, initial and reference conditions, and the various input and output file specifications. Additional discussion of the flow solver is presented in §5. Essentially, the run is initiated and following an initial transient the parameters that control advective and diffusive damping are adjusted to small values, the flow is advanced further to equilibrium, and then time-averaged if the run is unsteady. Relevant here is that this procedure generates the set of output files desired for visualizing the flow and if necessary, for grid adaption.

2.12 Adapting the grid

The region of interest is predicted or identified from the initial solution. This region, if adapted or refined, had been noted to yield better results (Pirzadeh (2001)). 'RefineMesh' was used for refining the grid. The project was read in and the region of interest was specified either by carving out a simple geometry such as a cuboid, cone, cylinder, etc. Also using the flow solution regions within the iso-surface of properties such as entropy, vorticity, pressure gradient, etc. could be carved out (Figure 10). This requires the solution in the .flo form. Once the region for refinement is chosen, the grid in this region was removed.

The grid density in the region is refined by reducing the ifact parameter in the d3m file. The change in ifact affects the intensity of the sources, and as explained earlier, affects the grid distribution locally. Then the grid in this region is regenerated using VGRID to obtain an adapted mesh.

3 Advancing Layers – Advancing Front Algorithm

3.1 Introduction

VGRIDns uses a hybrid advancing layers - advancing front technique for the generation of the grids. The advancing layers algorithm is based on a grid-marching strategy, resulting in the formation of high aspect ratio cells. Grids up to an aspect ratio of 20000:1 can be generated Pirzadeh (1993), compressed in the direction of the velocity gradient so as to capture the viscous fluxes as accurately as possible. The cells generated are extruded from the surface triangles and so form

prisms. These layers capture the boundary layer and cover the RANS region within a DES prediction. The cells in this layer are extruded from the surface grid until they meet the advancing layer from the opposite direction or reach the spacing criteria specified using sources by the user.

The advancing front technique is a boundary conforming technique, i.e, it starts from a boundary and advances until the whole domain is filled with cells. The cells formed using the advancing front technique are tetrahedral and are used to fill the inviscid region of the domain. The final layer formed by the advancing layers is used as the initial front for the advancing front algorithm. The grids are generated and the front updated after every cell is generated. A bounded volume is fully meshed until the front is empty. For an unbounded domain the grid is advanced until a set of user-defined conditions are arrived at. In the advancing front technique for unstructured grids the cell as well as the nodes are defined simultaneously unlike the Delaunay triangulation (or Tessalation) which is used to connect existing points.

3.2 Initial data

The geometry bounded by the domain forms the initial front for the algorithm. This is created in GridTool by dividing the domain into patches. The whole geometry is split into parts, each of which form a surface (planar, triangular or Coons patches). The boundaries of these form the initial front for the surface grid development.

The mesh parameters (such as the density of the mesh, the initial grid spacing, the stretching ratio, etc.) are controlled by the 'Background Grid' menu in Gridtool. The source strength and size are used to control the size of the grids while the "Global" sub-menu elements (delta, rate1, rate2) are used to control the viscous (prism) layers in the grids. The spatial distribution is given by the interpolation of the nodes from this data.

3.3 Advancing Layers algorithm

This is a systematic algorithm of generation comprised of stretching the cells into the domain from the surface. The layers are generated one at a time. This reduces the complexity of checking the cell connectivities, minimizes congestion of the cells and distributes the cells evenly so that the layers progressing towards each other do not overlap. The efficiency of this algorithm improves considerably in comparison to the advancing front technique, due to lesser complexity of the algorithm. The cell distribution is dependent on the stretching factor and the initial grid spacing provided by GridTool (delta, Rate1, and Rate2).

The cell distribution and the quality of the viscous grid is sensitive to the position of every point generated from the initial front. Thus, even a slight displacement of a point can affect the tetrahedral cells in the region just outside the prism grid layers. The points are positioned along the set of surface vectors (Figure 11) that are formed by averaging the normal unit vectors of the

15

faces sharing the point and then smoothing it using Laplacian smoothing. This facilitates better grid point distribution along the grid, even at points on the geometry with sharp corners. VGRIDns displays the surface vectors before the viscous layers are generated and the user can check the surface vectors before the prism layers are generated.

- To start with, new points are introduced in the field along the surface vectors and are connected to the faces from which they were generated.

- A list of faces is formed, and the faces are successively selected from the list and the suitable ones form a cell. Three types of faces (Figure 12) are formed as the layers advance:

  - Primary Faces: These are the faces which have all the nodes in the previously constructed layer.

  - Secondary Faces: These are the faces with one or more nodes (but not all) in the new layer, and at least one node in the previous layer. The nodes lie along two different surface vectors.

  - Cross-sectional Faces: These faces are similar to the secondary faces but both the nodes lie on the same surface vector.

The primary and the secondary layers are used to create cells, while the cross-sectional faces are removed from the front with a primary or secondary face.

- Since the selection of the points forming a cell can make a considerable impact on the quality of the cell the best cell has to be selected. Points to be considered apart from that formed using the extrusion along the surface vector include all the points from the opposite front. All of these are evaluated to form the best solution. A 'spring' analogy is used to select the point. The points are connected to the vertices of the face to be extruded, by means of tension springs. The resulting force exerted on a point is the sum of the tensions in each of the springs (which is proportional to the displacements). The point with the least force F is selected as the 'ideal' point (Figure 13).

$$F \alpha \sum_{n=1}^{N} d_n \tag{5}$$

where d_n is the displacement of the spring.

This enables the closest point to be the best candidate for the new point thus eliminating the need for face-crossing checks at later stages. Such a situation arises only when two approaching fronts meet. Normally the point on the surface vector forms the ideal point.

16

- After each layer (or cell) is formed then the old layer (cell) is removed and the new layer is considered as the front for the next layer.

- The generation is stopped when the front meets an opposite front or the size of the prisms approaches the size prescribed by the sources (Figure 9).

The final layer generated forms the initial front for the Advancing Front Algorithm.

3.4 Advancing Front algorithm

The advancing front algorithm Peraire *et al.* (1998), Hoffman and Chang (1998) is an algorithm in which the cells and the points are formed simultaneously. The algorithm is summarized as follows:

- To start with, the domain boundaries are initially divided into a set of control points. The surface grid generation can proceed once the boundaries are discretized as the discretized boundary forms the initial front.

- If the grid is viscous, then the prism boundary layers are generated using the Advancing Layers algorithm and the outer most grid layer is considered as the initial front for the Advancing Front algorithm.

- To start the grid generation process a side from the front is chosen. Usually the algorithm is started with the smallest side. This process, of selecting the smallest side, becomes faster and more efficient if proper data structures are used.

- With the details in the 'Background grid' the mesh parameters are interpolated at the center of the side chosen. Using these details the "ideal" node, C, is selected in the domain. The ideal node (in 2D) forms an isosceles triangle with the end points (A or B) of the side chosen (Figure 14). The distance, δ_1, from the end point to the new node C is given by

$$\delta_1 =$$

  - 1.00 — if $0.55L < 1.00 < 2.00L$

  - $0.55L$ — if $0.55L < 1.00$

  - $2.00L$ — if $1.00 > 2.00L$

  - where L is the distance between the two end points.

These coefficients are empirical and different inequalities also have been used Peraire *et al.* (1998), Hoffman and Chang (1998). This is to ensure that no distorted cells are produced.

- To avoid poor quality triangles (tetrahedra) in the later stages the nodes in the vicinity of the node (C) within the radius δ_1 are to be found (Figure 14). This can be made faster and more efficient by the use of suitable data structures (like Alternating Digital Tree or quad-tree/Octree). This radius is also an empirical value and other values can be used to increase the robustness of the algorithm.

- The nodes in the region (Q1, Q2) are listed in the order of the distance from the initial points A,B and the points C, C1, C2, ... (that lie on the perpendicular bisector of the side AB) are to be appended to the list. The points C1, C2, ... are necessary so as to ensure that some triangle (tetrahedra) is formed if not the ideal node.

- The best connecting point is selected from the list. This is the first point in the list which does not intersect with the currently existing front.

- Once this node is formed the front is updated and the algorithm repeated until the whole surface (volume) mesh is generated (c.f., Figure 15).

The volume can be filled with either prism layers or tetrahedron. The prism layers are extrusion of the surface grid generated in the earlier stage, while the tetrahedral grids are generated using the surface grid or the outer most prism layer as the initial front and proceeding on with the 3D advancing front algorithm. The volume grid generation is very similar to the surface grid generation. The points C1, C2, ... lie on the centroid of the starting triangle in this case and δ_1 is calculated by taking the average of the three sides of the starting triangle as L.

The 3D advancing front algorithm can be observed in VGRIDns by choosing the grid generation by 'Fly' (Figure 15). This shows the updated grid after the specified iteration intervals.

3.5 Mesh quality improvement

To enhance the quality of the mesh generated, post-processing procedures are applied. This is done in Postgrid in the present effort. The procedures are more local in nature and so do not alter the total number of cells present in the grid. The pockets left out in VGRIDns are remeshed in Postgrid and negative and distorted cells can be removed and replaced by cells of better quality.

1. Diagonal swapping: This process swaps the diagonals in two neighboring cells to form a pair of new cells. This does not alter the position and can be carried out in all the points that do not lie on the boundary. The swapping (Figure 16) is performed only if the new configuration applies better than the previous one (i.e., in this case the distortion in the cell becomes less or a cell with a better aspect ratio is formed).

2. Mesh smoothing: This process involves altering the position of the interior nodes (Figure 17). The nodes are considered to be sets of springs and the nodes are moved such that an equilibrium is reached in the system. This is carried over a set of 3 to 5 iterations.

A combination of the above two processes can be used to generate a fine mesh without any negative (overlapped) or flat cells.

3.6 Performance of the algorithm

The performance of the algorithm depends on the grid size, domain shape and the complexity of the mesh parameters. The performance improves with the effective use of data structures, since most of the time is spent in searching for a node, face or a cell. A simple check of intersection of faces which can take up to 80% of the grid generation time can be reduced to 25% by use of suitable data structures. As mentioned above the Alternating Digital tree technique and the quad-tree/oct-ree technique are used to enhance the performance of the algorithm

4 Turbulence Modelling Strategies

Current turbulence simulation strategies range from Direct Numerical Simulations (DNS) to Reynolds-averaged Navier Stokes (RANS). While in some strategies empiricism is an essential aspect of the modeling, in others there is less or no empirical input (as in the case of DNS). Grid density, for example, plays a role in determining the resolution capacity of strategies such as LES and DES. The tradeoff is the usual one – lowering empirical input increases the computational cost of the approach. These competing factors - accuracy and computational efficiency – are important in dictating the selection of a turbulence model and an optimization of both quantities is a worthwhile goal.

4.1 Direct Numerical Simulation

Direct Numerical Simulation (DNS) is an approach which resolves all the turbulent scales. This approach is free from any empiricism as DNS solves the Navier-Stokes equation without any modeling or averaging. DNS provides very detailed information of turbulence which can be difficult to obtain using experiments. As mentioned in Spalart (2000) one run of thorough DNS is equivalent to three of our LES predictions since a DNS is stand-alone.

As is well known, the range of scales in a turbulent flow is a function of the Reynolds number. Energy supplied to the largest eddies (of length scale L) by the mean flow is transfered to the smaller scales continuously until the energy is dissipated into heat by the smallest eddies, which are of dimension corresponding to the Kolmogorov length scale (η). The length of the largest eddies are of the order of computational domain. The smallest eddies on the other hand determine

19

the minimum grid spacing required in DNS. Thus, the grid size in each direction is dependent on the ratio between the largest and the smallest length scales ($L/\eta \propto Re^{3/4}$). This gives the Reynolds-number scaling of the grid for DNS as $\propto Re^{9/4}$. The time scales are bound by the ratio of the integral time scales and the Kolmogorov time scales which is proportional to $Re^{3/4}$. Since the CPU time is dependent on both the grid size and the number of time steps required, the total cost depends on the cube of the Reynolds number. This computational cost limits the application of DNS to moderate Reynolds number regimes.

4.2 Large-Eddy Simulation

Large-Eddy Simulations (LES) is a compromise between DNS in that it introduces empiricism to model small-scale motions but resolves the large, energy-containing scales of motion. In LES the flow is decomposed by means of a filter into small and large length scale contributions with the larger scales being resolved. The cost of the computation is lower than in DNS and LES is especially powerful away from solid surfaces where the Reynolds number scaling is very favorable. Thus, LES can be used in certain flow regimes where DNS is not possible (some high Reynolds numbers flows, more complex geometries, etc.).

4.2.1 Subgrid scale modeling

Most subgrid scale models takes advantage of the fact that the global dissipation level is set by the largest eddies and the smaller eddies exhibit similar behavior while transmitting energy to the smallest scales of motion. In LES, the small scales whose effect on the large eddies must be modeled are removed from the flow using a low-pass filtering operation,

$$\bar{f}(\vec{x}) = \int f(x')G(\vec{x} - \vec{x'}; \Delta)dx' \tag{6}$$

where the filter function is G and the filter width is Δ. The filtered Navier-Stokes equation can be written as

$$\frac{\delta \bar{u}_i}{\delta t} + \frac{\delta}{\delta x_j}(\bar{u}_i \bar{u}_j) = -\frac{1}{\rho}\frac{\delta \bar{p}}{\delta x_i} - \frac{\delta \tau_{ij}}{\delta x_j} + \nu \frac{\delta^2 \bar{u}_i}{\delta x_j \delta x_j}. \tag{7}$$

The subgrid scales influence the flow through the subgrid scale stress,

$$\tau_{ij} = \overline{u_i u_j} - \bar{u}_i \bar{u}_j \tag{8}$$

which has to be modeled for closure.

Because the dissipative scales in LES are resolved poorly or not resolved at all, the energy associated with the resolved scales is removed by the SGS models and act as a drain associated with the energy cascade. Most of these models are eddy viscosity models of the form,

$$\tau_{ij} - \delta_{ij}/3\tau_{kk} = -2\nu_t \bar{S}_{ij} = \nu_t \left(\frac{\delta \bar{u}_i}{\delta x_j} + \frac{\delta \bar{u}_j}{\delta x_i}\right), \tag{9}$$

20

relating the subgrid scale stresses to the large scale strain rate tensor, $\bar{S}_{ij}$. In most of the cases the small scales are assumed to be in equilibrium, i.e, the energy obtained from larger scales is dissipated instantaneously and completely to smaller scales. An algebraic relation for the eddy viscosity is obtained as,

$$\nu_t = c\Delta^2|\bar{S}|\bar{S}_{ij} \qquad |\bar{S}| = (2\bar{S}_{ij}\bar{S}_{ij})^{\frac{1}{2}}. \tag{10}$$

This model is the "Smagorinsky model" and the constant c is usually determined via simulations of isotropic turbulence.

4.3 Reynolds-averaged Navier-Stokes Approaches

In RANS approaches, flow variables are decomposed into a mean and a fluctuating component,

$$F(x,t) = \overline{F}(x) + F'(x,t), \tag{11}$$

where the above relation implies a time average for a statistically stationary flow, i.e.,

$$\bar{f}(\vec{x}) = \lim_{T \to \infty} \frac{1}{T} \int_t^{t+T} f(x',\tau)d\tau \tag{12}$$

When the average is applied to the Navier-Stokes equation, equations for the mean flow are obtained, with the turbulent motions appearing through the Reynolds stress,

$$\tau = -\rho\overline{u_i'u_j'} \tag{13}$$

This is the unknown quantity which has to be modelled and in the majority of approaches is closed using eddy viscosity. In the present effort, the Spalart-Allmaras (referred to as S-A throughout) model has been used for modeling the Reynolds stress (via solution of a transport equation for the eddy viscosity).

4.3.1 Spalart-Allmaras one equation model

The Spalart-Allmaras turbulence model Spalart-Allmaras (1997) is derived by an amalgamation of empiricism, dimensional analysis, Galilean invariance and selective dependence of molecular viscosity. One of the practical advantages of the model is that it is compatible with grids of any topology. As pointed out in Spalart-Allmaras (1997), some of the earlier turbulence models (Baldwin-Lomax, Cebeci-Smith, Johnson-King) relied on velocity or vorticity profile varying along a smooth gridline and so were non-local but restricted to structured grids. Some other transport models (such as k-ϵ) are local but lack accuracy in prediction of shock/boundary layer interactions or separations on smooth surfaces. The Baldwin-Barth one equation model is local and does not require fine resolution of the grid at the walls. This can be advantageous in terms of accuracy and practicality compared to two-equation models. S-A model has loose connections

to the Baldwin-Barth model, which is derived from the k-ϵ model with some assumptions and so this dependence on the k-ϵ model puts constraints on the diffusion term, for example. The S-A model is more robust than Baldwin-Barth and has similar properties in terms of compatibility with unstructured grids and near wall behavior.

The S-A model is developed over four "versions" with the simplest one applicable to free shear flows and the most complicated one to viscous flows past solid bodies with laminar regions. The S-A turbulence model solves an equation for the variable $\tilde{\nu}$ which is dependent on the turbulent viscosity. This model includes a wall destruction term that reduces the turbulent viscosity in the laminar sub-layer and trip terms to provide smooth transition to turbulence. The differential equation for this model is written as,

$$\frac{D\tilde{\nu}}{Dt} = C_{b1}[1 - f_{t2}]\tilde{S}\tilde{\nu} + \frac{1}{\sigma}[\nabla.((\nu + \tilde{\nu})\nabla\tilde{\nu}) + c_{b2}(\nabla\tilde{\nu})^2] - [c_{w1}f_w - \frac{C_{b1}}{\kappa^2}f_{t2}][\frac{\tilde{\nu}}{d}]^2 + f_{t1}\triangle U^2 , \quad (14)$$

with the trip functions being expressed as,

$$f_{t1} = c_{t1}g_t\exp\left(-c_{t2}\frac{\omega_t^2}{\triangle U^2}[d^2 + g_t^2d_t^2]\right) , \qquad g_t = \min\left(0.1, \frac{\triangle U}{\omega_t \triangle x_t}\right) \quad (15)$$

where $\triangle x_t$ is the grid spacing along the wall at the trip, d_t is the distance from the field point on the trip (or the wall), ω_t is the wall velocity at the trip, and $\triangle$ is the difference in the velocities at the trip and the field point. The second trip function is expressed as,

$$f_{t2} = c_{t3}e^{-c_{t4}\chi^2} , \quad (16)$$

where the turbulent viscosity is calculated as,

$$\nu_t = \tilde{\nu}f_{v1} , \qquad \frac{\chi^3}{\chi^3 + c_{v1}^3} , \qquad \chi = \frac{\tilde{\nu}}{\nu} , \quad (17)$$

where the modified vorticity is,

$$\tilde{S} = S + \frac{\tilde{\nu}}{\kappa^2 d^2}f_{v2} , \quad (18)$$

where S is the magnitude of vorticity, d is the distance to the closest wall, and,

$$f_{v2} = 1 - \frac{\chi}{1 + \chi f_{v1}} . \quad (19)$$

The wall destruction function f_w given by,

$$f_w = g[\frac{1 + c_{w3}^6}{g^6 + c_{w3}^3}]^{\frac{1}{6}} , \qquad g = r + c_{w2}(r^6 - r) , \qquad r = \frac{\tilde{\nu}}{\tilde{S}\kappa^2 d^2} . \quad (20)$$

The model coefficients are given by $c_{b1} = 0.1355$, $\sigma = 2/3$, $c_{b2} = 0.622$, $\kappa = 0.41$, $c_{w1} = c_{b1}/\kappa^2 + (1 + c_{b2})/\sigma$, $c_{w2} = 0.3$, $c_{w3} = 2$, $c_{v1} = 7.1$, $c_{t1} = 1$, $c_{t2} = 2$, $c_{t3} = 1.1$, $c_{t4} = 2$.

4.4 Detached-Eddy Simulation

DES is a hybrid RANS/LES model and was originally proposed by Spalart (1997). In natural applications, essentially the whole boundary layer relies on a RANS treatment while the separated regions ('detached eddies') are treated using LES. This is affordable even at high Reynolds numbers and has been validated by various groups, e.g., see Squires $et\ al.$ (2002), Travin$et\ al.$ (2000), Spalart (2001). The flow around a whole aircraft has been simulated using DES Forsythe $et\ al.$ (2002), while such a simulation using DNS or LES seems difficult, if not impossible in the near future.

DES requires directing the grid resolution in desired areas, which demands some skill from the user. The grid parameters in the streamwise and the spanwise directions can be relaxed as compared to LES but the wall normal spacing in the grid should be of the order of $y^+ = 1$. The grid requirements are described in detail in the next chapter.

DES is not a zonal technique and so the problems of identifying the area governed by the models using 'artificial intelligence' is not required. This makes it preferable for routine use as it requires just a small alteration of the RANS model. DES uses a switch to cross over from RANS to LES and so the simplest type of RANS models that make DES practical are one equation models. Most of the simulations used so far have been carried out using the Spalart-Allmaras model. DES is also feasible on both structured Shur $et\ al.$ (1999) as well as on unstructured grids Forsythe (2000), Squires $et\ al.$ (2002), Squires $et\ al.$ (2002).

The DES formulation used here is based on a modification to the Spalart-Allmaras RANS model Spalart-Allmaras (1997) such that the model reduces to its RANS formulation near solid surfaces and to a sub-grid model away from the wall Spalart (1997). In DES, the distance to the nearest wall d is replaced by $\tilde{d}$,

$$\tilde{d} \equiv min(d, C_{DES}\Delta)\,, \tag{21}$$

where Δ is the maximum cell dimension in the three dimensions. For an unstructured grid Forsythe (2000) redefined Δ as the largest inter-centroidal distance. In "natural" applications of DES, the wall-parallel grid spacings (e.g., streamwise and spanwise) are on the order of the boundary layer thickness and the S-A RANS model is retained throughout the boundary layer, i.e., $\tilde{d} = d$. Consequently, prediction of boundary layer separation is determined in the "RANS mode" of DES. Away from solid boundaries, the closure is a one-equation model for the sub-grid scale eddy viscosity. When the production and destruction terms of the model are balanced, the length scale $\tilde{d} = C_{DES}\Delta$ in the LES region yields a Smagorinsky-like eddy viscosity $\nu_t \propto S\Delta^2$. Analogous to classical LES, the role of Δ is to allow the energy cascade down to the grid size. The additional model constant $C_{DES} = 0.65$ was set in homogeneous turbulence (Shur $et\ al.$ (1999)).

4.5 Detached-Eddy Simulation grids

Grid design poses a challenge for any turbulence-resolving strategy such as DES. The RANS and LES regions are dictated by the grid spacings and an inappropriate grid can defeat the usefulness of DES. Since flow fields with widely different gridding requirements are usually encountered, a generalized methodology to generate DES grids was proposed by Spalart (2001).

The regions in a DES case can be broadly divided into the Euler regions, RANS region, and LES region based on the grid spacings. These regions are termed as super-regions or parent regions by Spalart (2001). The RANS region is sub-divided into a viscous region and outer region while the LES region consists of a viscous region, focus region, and the departure region (Figure 19). A summary of these regions, borrowing heavily from Spalart (2001) is presented next.

4.5.1 Euler Region

The Euler region is the part of the flow field which is rarely affected by turbulence or vorticity generated. This region covers most of the volume in the computational domain but due to its relative insignificance it constitutes a smaller share of grid points. Using an unstructured grid, this region is filled by tetrahedral cells, growing from the sources placed on the corners of the domain.

4.5.2 RANS Region

This region is primarily the boundary layer including shallow separation bubbles. The region is usually simulated in the RANS mode, but a very fine grid in this region activates the LES mode. This region is of importance and constitutes of a large fraction of the total number of grid cells. This region can be further sub-divided as follows,

RANS Viscous Region

This is the region closest to the solid wall. Since this region is entrusted to the RANS mode of the modeling, the grid requirements are similar to those for a standard RANS application. This region accounts for the modeled portion of the stresses in the simulation and covers the viscous sub-layer, buffer layer, and the log layer. Due to the relative importance of the layer it constitutes a majority of the grid cells. The first wall normal spacing is determined by RANS and so the restriction of the first wall normal spacing is typically within one viscous unit. However, the spanwise and the streamwise spacings do not have such restrictions, unlike the LES or DNS case which demands tighter spacings in these directions. However in case of geometrical discontinuities the spacing in these directions should scale with the variation of the geometry so as to capture the flow variations in these areas. A stretching ratio of around 1.25 is desired to accurately predict the log layer. Experience shows that going below a stretching ratio of 1.2 and a $y^+ = 1$, has little advantage (Spalart (2001)).

24

In VGRID this region is represented using prisms which are extruded from the surface grids. The spacings parallel to the wall are dependent on the surface grids while the wall normal distances and the stretching ratios are set using the delta, Rate1, and Rate2 terms. Since the source size determines the extent to which the prism grids would grow, the source sizes should be fixed so that the RANS region lies within the prism layers.

RANS Outer Region

The RANS outer region covers the outer regions of the boundary layer. This is the approximate region where the velocity gradients normal to the wall are not as sharp as it is near the wall. The eddy viscosities however have steep gradients in this region. The grids should enable a smooth transition of eddy viscosity from this modeled region to the resolved LES region. Therefore, Spalart (2001) suggests that over-predicting the boundary layer thickness (or fixing the outer edge of the prism layers in VGRID), is safer rather than under-predicting it. This ensures that the solution inside the boundary layer does not feel the effect of the grid spacing. The solver should also be able to tolerate the slope discontinuity for accurate prediction of the flow.

This forms the outer region of the prism layers and by over-predicting the boundary layer this region can also be accommodated inside this layer. However, the smooth transition of the grid from prisms to tetrahedron by combining the intermediate regions into pyramids, in Blacksmith, helps in the smooth variation of the eddy viscosity, in this region if the boundary layer has been slightly under-predicted.

A good estimate of the RANS region can be made for setting the grid parameters. Since the RANS-LES interface between the regions is a grid dependent parameter, an approximate estimate of the parameters (source strength, growth rate) can be made. A comparison between the distance of the cell centroid from the wall and the grid dimensions can be made for a specific set of parameters and the interface approximately located. An approximate evaluation for a spheroid case is shown in Figure 18.

4.5.3 LES Region

This is the region of interest in most cases, especially for flows experiencing massive separation. This involves the regions in the wake where the flow encounters significant unsteadiness turbulence, vorticity, etc. The purpose of grid refinement in this region is to obtain richer physics. Isotropy in the LES region grids is advantageous and an unstructured grid assures near-isotropy in this region. This region is further sub-divided as follows:

LES Viscous Region

This is the region in proximity to the wall, but in the separated areas of the flow. The grid requirements are the same as the RANS viscous region, and so the prism layers cover this region in the unstructured grids applied in this work.

25

Focus Region

The focus region is the region where the possibility of a particle returning to a point very close to the wall (or to some point of interest in the flow) is high. This is the primary area of interest in most of the simulations and the grid should be constructed such that it resolves the eddies well in this region. The grid size is the primary measure of the spatial resolution in a DES simulation, and should be close to the principal spatial resolution applied in classical LES. Adaption should focus on this region and the grid can be refined by carving out this region in RefineMesh and redefining the 'ifact' value in the d3m file accordingly. Since this region approximately lies outside the prism layers, the size of the grid cells is on the order of the size of the sources defined by the user in GridTool on or near the body. Thus, the user has the capability to control the size of the grids in this region. The grid sizes increases as the distance from the wall increases, the Focus Region then slowly merging with the departure region. This variation can also be controlled with the help of the source strengths (*an,bn*) in GridTool. However, even if the Focus Region does not merge into the departure region and goes directly into the Euler region or to the outflow boundary, there is not much difference is noted in the solution quality, at least on the present grids and domains.

Departure Region

This is the region lying in the wake which usually connects the Focus Region and the Euler region. This region is not in direct contact with the surface but is affected by the presence of the body. The grids in this region are coarse, and this region is associated with large scale dynamics. Fixing the intermediate zones between the regions is the main objective of the grid generation process. Sharp boundaries often do not exist between the regions outlined above. The boundary between the Euler region and any other region should be free from the effects of the body. Thus, in cases such as a transition from a RANS region to an Euler region, there can be sudden changes in the grid sizes. In the region of transition from the focus region to the departure region, coarsening of the grids can be also expected.

Thus, the main objective of the user is to generate a grid that is fine enough to recognize the transformation of the outlined regions in the flow. As Spalart (2001) quotes, any unsatisfactory results are a result of the user's failure to create an acceptable grid.

4.6 Effects of grid refinement

Grid refinement serves different purposes in the various approaches summarized above. For a DNS or LES approach the basic function of grid refinement is to obtain richer turbulence physics. Refined grids resolve the smaller eddies and thus the larger eddies have more eddies for non-linear interaction, making the solution more accurate. However while increasing mesh resolution improves accuracy in these approaches, it is precisely this requirement that makes the approaches computationally expensive.

On the contrary, in RANS models grid refinement is aimed at a smooth exact solution of the governing equations. The aim of grid refinement is then numerical rather than physical. Since the basic equations solved in RANS models are empirical relations, grid refinement merely provides a better discretization of the solution. Beyond a certain limit, further grid refinement has little effect on the solution and convergence is achieved.

DES case being a hybrid approach shows a more complicated interaction with the grid. While the LES region of the solution will benefit from refinement, an overall refinement may result in a loss of accuracy if the RANS-LES interface is located too close to the wall. The interface is a grid dependent parameter and so it is fixed initially in the solution. The model is designed such that a major portion of the boundary layer is modelled in RANS mode. The switch to LES mode takes place when the distance of the wall from the grid centroid is of the order of the distance of the farthest neighbor (a cell sharing a face). The relation being given by $\tilde{d} = C_{DES}\Delta$, and the factor C_{DES} fixed at 0.65. The streamwise or the spanwise spacings are almost always larger than the wall normal grid spacing in the boundary layer, and so these spacings fix the interface. A fine grid could activate the DES limiter within the boundary layer, resulting in the decrease in the modelled eddy viscosity. An interface well inside the boundary layer could then predict earlier separation of the flow, due to the lower eddy viscosity levels. Thus, a DES grid should not only be fine enough to capture the variations in the LES Region, but also appropriate to the RANS Region with the interface outside of the boundary layer.

Since in unstructured grids, generated using the advancing layers-advancing front technique, the transformation from the prism layers to tetrahedron takes place when the aspect ratio of the cell approaches unity, the interface should lie inside the prism layers. Refinement in this region generally leads to adverse effects on the solution. Grid refinement in the outer region, where more isotropic cells are present, improve the LES characteristics of the solution via capturing the physics. This is the region where 'Refinemesh' is applied, thus leaving the prism layers and the RANS-LES interface intact.

5 Numerical Simulation using *Cobalt*

Cobalt is a cell-centered finite volume approach based on Godunov's first-order accurate, exact Riemann method. The solver is second order accurate in space and time and has implicit time stepping. To the baseline numerical method, the viscous terms and turbulence models are added. *Cobalt* has been developed at the Interdisciplinary and Applied CFD section of the Wright-Patterson Laboratories. *Cobalt* has undergone several improvements since it was introduced in 1996. The algorithm followed consists of five fundamental tasks: (a) construction of initial conditions for the Riemann problem (Hirsch (1990)) at a given cell face, (b) solution of the Riemann problem, (c) Construction of the viscous fluxes at a given face, (d) time integration, and (e) impo-

sition of boundary conditions. The development in this is taken largely from Strang *et al.* (1999).

This governing equations being solved can be written in the form,

$$\frac{\partial}{\partial t}\int_V Q dV + \int_S (E\hat{i} + F\hat{j} + G\hat{k}).\hat{n}dS = \int_S (E_v\hat{i} + F_v\hat{j} + G_v\hat{k}).\hat{n}dS \qquad (22)$$

where V is the element volume, S is the surface area, $\hat{n}$ is the outward unit normal to S, and $\hat{i}, \hat{j}, \hat{k}$ are the Cartesian unit vectors. The solution vector contains,

$$Q = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho e \end{bmatrix}, \ E = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uw \\ u(\rho e + p) \end{bmatrix}, \ F = \begin{bmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ \rho vw \\ v(\rho e + p) \end{bmatrix}, \ G = \begin{bmatrix} \rho w \\ \rho uw \\ \rho vw \\ \rho w^2 + p \\ w(\rho e + p) \end{bmatrix} \qquad (23)$$

and,

$$E_v = \begin{bmatrix} 0 \\ \tau_{xx} \\ \tau_{xy} \\ \tau_{xz} \\ u\tau_{xx} + v\tau_{xy} + w\tau_{xz} + kT_x \end{bmatrix}, \ F_v = \begin{bmatrix} 0 \\ \tau_{xy} \\ \tau_{yy} \\ \tau_{yz} \\ u\tau_{xy} + v\tau_{yy} + w\tau_{yz} + kT_y \end{bmatrix}, \qquad (24)$$

$$G_v = \begin{bmatrix} 0 \\ \tau_{xz} \\ \tau_{yz} \\ \tau_{zz} \\ u\tau_{zx} + v\tau_{zy} + w\tau_{zz} + kT_z \end{bmatrix} \qquad (25)$$

where T is the temperature, e_t is the specific energy/volume, and τ are the shear stresses.

The equations in semi-discrete form can be written as,

$$V_i\frac{dq_i}{dt} + \sum_{m=1}^{N_i}(E_m\hat{i} + F_m\hat{j} + G_m\hat{k}).\hat{n}_m S_m = \sum_{m=1}^{N_i}(E_{vm}\hat{i} + F_{vm}\hat{j} + G_{vm}\hat{k}).\hat{n}_m S_m \ . \qquad (26)$$

The ideal gas law then closes the system of equations along with the turbulence models that are decoupled from the main conservation equations.

5.1 Inviscid and viscous fluxes

5.1.1 Inviscid fluxes

The inviscid fluxes are determined by assuming a linear variation of the solution in each cell. This yields second-order spatial accuracy to the method. The initial condition to the Riemann problem can be formulated as,

$$q^f = q_i + \vec{r}_f \cdot \nabla q_i \ , \qquad (27)$$

where q^f is the value at the face centroid, $\bigtriangledown q_i$ is the gradient vector for cell i, $\vec{r}_f$ is the vector from the centroid to the face. A centered approximation is calculated using the properties in the nearest-neighbor cells. This is expressed as,

$$A\bigtriangledown^c q_i = \sum_{m=1}^{N_i} (q_m - q_i).$$ (28)

The unknown value at the face centroid is replaced by the values at the opposing cell centroids. The relation matrix A is given by these known values of cell centroids. This matrix is solved using QR factorization producing a least-squares result. The QR factorization is more stable than an LU decomposition and for high aspect ratio cells produced in the boundary layer by VGRID, the stability is of importance. The relation reduces to,

$$\bigtriangledown^c q_i = \sum_{m=1}^{N_i} \omega_i^{\vec{m}} (q_m - q_i)$$ (29)

where $\omega_i^{\vec{m}}$ is a grid-related quantity and so is fixed throughout the run. The central difference gradient is used for extremal values at the cell centroids. In case of a non-extremal case a one sided difference least squares approach is used. Since least square is used the error in the fit is minimum. A Total Variation Diminishing (TVD) scheme are used to limit the extremes at the cell faces.

5.1.2 Total Variation Diminishing (TVD) scheme

The algorithm used to limit quantities is based on the idea of switching to one-sided differencing in regions of extrema. The left and right values of a given fluid quantity are computed for each face and the cell producing an extrema is flagged. A one-sided difference is used to compute the properties for each of the flagged cells. If the computed value still remains extremal, then the value is limited by resetting it to the highest of the left and right cell centroid values.

A one-sided least-squares gradient can be obtained by correcting the original central-difference, least squares gradient. A correction is made by adjusting the state of the flagged cell so as to minimize the error in least squares sense,

$$\frac{\delta}{\delta q_j} \sum_m^U e_m^2 = 0$$ (30)

where

$$e_m = q_m - (q_i + r_i^{\vec{m}}.\bigtriangledown^c \cdot q_i)$$ (31)

where U is the set of all flagged cells, e_m is the error from the fit at all the neighboring cells using the central difference, and the vector $r_i^{\vec{m}}$ is the vector connecting the centroids of the cell with the neighboring cell centroid.

This results in a one-sided least squares gradient given by,

$$\vec{\nabla}^c q_i + \vec{\omega}_i^j \frac{\sum_m^U e_m (\vec{r}_i^m . \vec{\omega}_i^j)}{\sum_m^U (\vec{r}_i^m . \vec{\omega}_i^j)^2}, \tag{32}$$

which is a TVD method. However if the nearest-neighbor cells fail to produce a change in the values, then such sets of U are considered as "incomplete". This can be rectified by eliminating the contribution of the flagged cells from the central difference fluxes before applying the corrections.

The Riemann method uses a combination of the approximate Riemann method of Colella (1982) and Newton's procedure outlined in Gothlieb and Groth (1988). The procedure of Gottlieb and Groth converges faster than the method suggested by van Leer (1982).

5.1.3 Viscous fluxes

With the eddy viscosity ratios (eddy viscosity/molecular viscosity) in some regions of the grid having low values, the viscous fluxes dominate the inviscid fluxes in many regions of the flow. Viscous fluxes are generally able to enhance the stability of the flow solver. These fluxes are computationally expensive and can account for a relatively large fraction of the computational overhead. However, their calculation is worth the effort due to the reliability and convergence properties, also essential in viscous flows. The formulation of the viscous terms should be conservative, must satisfy discrete maximum principle (Barth (1991)), and the null result should be returned for $\nabla \cdot \nabla q$ on a linear field. Conservation can be easily satisfied by constructing viscous fluxes for the faces, but simultaneously satisfying both the other conditions is difficult. To accurately capture boundary layers, cells are compressed in the direction of the velocity gradient. The resulting vector joining cell centroids is normal to the velocity gradient. This gives reasonable results as it satisfies the discrete maximum principle condition, apart from being conservative. Although the $\nabla \cdot \nabla q$ does not vanish for a linear q, the results can be reasonable.

5.2 Riemann problem

The Riemann problem was introduced in the field of Computational Fluid Mechanics in the late 50s by Godunov. This formulation forms the core of *Cobalt*. The problem was used by Glimm *et al.* (1965) in conjunction with random sampling methods to derive proofs for conservation laws.

The initial value problem with initial value,

$$U(x, t_j) = U_i, x < x_i + \Omega(X_i - x_{i+1}), \tag{33}$$

$$U(x, t_j) = U_{i+1}, x \geq x_i + \Omega(X_i - x_{i+1}). \tag{34}$$

With time the two states break into leftward and rightward moving waves and depending on the initial conditions form wave patterns which are either shocks or rarefaction waves. The de-

termination of the type of waves generated, their properties and strengths for a given set of initial conditions is referred to as the Riemann problem and the algorithm is called the Riemann solver.

For a one-dimensional case the variables pressure, density and the velocity of the wave on either sides of the interface is determined. The conservation equations reduce to Rankine-Hugoniot equations for a shock wave and an isentropic characteristic equation across rarefaction waves. These equations are used to jump across the moving waves into the unknown state that exists between the two waves generated in the left and the right side. Assuming a single state to exist between the waves, on both the right and left sides, the equations reduce to a single non-linear algebraic equation in one unknown for the wave pattern. This is implicit in pressure or velocity in this region and can be solved iteratively.

An efficient solver should have the fewest number of mathematical operations for the entire solution procedure. The initial condition, the equations used in the iterative procedure, the check to stop the iterations, and expressions to determine the state in the left and right of the contact surface affect the computational efficiency of the algorithm.

5.3 Riemann solver: Gottlieb and Groth

Gothlieb and Groth (1988) proposed a solver in which the set of variables (p, c, u, γ, R) are used instead of (p, ρ, u, γ, R) as the speed of sound appears more frequently. The intermediate flow velocity (u^*) was used instead of the pressure (p^*) to iterate and the pressures across the contact made equal $(p_l^* = p_r^*)$. Newton's iterative method is used to calculate u^* using,

$$u_{i+1}^* = u_i^* \frac{p_l^*(u_i^*) - p_r^*(u_i^*)}{p_l^{*\prime}(u_i^*) - p_r^{*\prime}(p_i^*)},$$ (35)

with convergence given by relations such as,

$$|1 - \frac{p_l^*}{p_r^*}| < \epsilon \sim 1 \times 10^{-6}.$$ (36)

The values p_l^*, p_r^*, c_l^*, c_r^*, dp_l^*/du^* and dp_r^*/du^* are calculated from the shock and the rarefaction wave equations. The shock equations ($u^* \leq u$, u is u_l or u_r) are,

$$W = \frac{\gamma + 1}{4} \frac{u^* - u}{c} - (1 + (\frac{\gamma + 1}{4} \frac{u^* - u}{c})^2)^{\frac{1}{2}}$$ (37)

$$p^* = p + C(u^* - u)W, \qquad p^{*\prime} = \frac{2CW^3}{(1 + W^2)}$$ (38)

$$c^* = c\frac{(\gamma + 1) + (\gamma - 1)p^*/p}{(\gamma + 1) + (\gamma - 1)p/p^*}, \qquad C = \frac{\gamma p}{c}$$ (39)

where W is the shock Mach number with respect to the gas moving ahead of the shock (W is a by-product of iteration). The shock velocity is $V = u + cW$. The corresponding rarefaction

31

equations are

$$c^* = c - \frac{\gamma - 1}{2}(u^* - u), \qquad p^* = p(\frac{c^*}{c})^{\frac{2\gamma}{\gamma-1}}, \qquad p^{*\prime} = \gamma\frac{p^*}{c^*} \qquad (40)$$

where c^* is a by-product of the iterations, $u + c$ is the velocity of the expansion wave head in the laboratory frame of reference. The initial guess to the solution is written as,

$$u_o^* = \frac{\bar{u}_l z + \bar{u}_r}{1 + z}, \qquad \bar{u}_l = u_l + \frac{2}{\gamma - 1}c_l, \qquad \bar{u}_r = u_r + \frac{2}{\gamma - 1}c_r \qquad (41)$$

$$z = \frac{\gamma_l - 1}{\gamma_r - 1}\frac{a_r}{a_l}(\frac{p_l}{p_r})^{\frac{\sigma+1}{\sigma}} \qquad (42)$$

$$\sigma = \gamma_r p_l < p_r, \qquad \text{or} \qquad \sigma = \gamma_l p_l \geq p_r \qquad (43)$$

This initial guess is accurate for wave patterns with shocks and superior to any initial guess based on acoustic theory. The velocity (u^*) is used instead of pressure (p^*) as this gives better results. Also the pressure based predictions are inaccurate in case of strong shocks, where the velocity predictions are inaccurate.

5.4 Glimm's method

Glimm's method is based on the concept that a piecewise continuous flow can be represented by a series of discontinuities ($\sim O(\Delta x)$), where the spatial increment in the grid is Δx. These waves have different strengths and propagate with different speeds. The strengths and the speeds change in a continuous fashion, varying im time. These discontinuities are modeled as a series of Riemann problems, separated by jumps. Conditions are defined to avoid the interaction between the adjacent waves. Thus the solution of these problems gives a solution close to the exact solution of the problem.

5.5 Temporal integration

In this algorithm the governing equations are recast as,

$$\theta(V\frac{\delta Q}{\delta t} + \vec{\nabla}.\vec{f})_i^{n+1} + (1 - \theta)(V\frac{\delta Q}{\delta t} + \vec{\nabla}.\vec{f})_i^n = 0 \qquad (44)$$

where $\vec{f}$ is the flux vector, $n, n+1$ represent successive time levels, θ is the temporal characteristics ($\theta = 0$ for an explicit method and $\theta = 1$ for an implicit method). The temporal derivatives expressed in discrete form are,

$$(\frac{\delta Q}{\delta t})^{n+1} = \frac{\alpha_{1,1}(Q^{n+1} - Q^n) + \alpha_{1,2}(Q^n - Q^{n-1})}{\Delta t} \qquad (45)$$

$$(\frac{\delta Q}{\delta t})^n = \frac{\alpha_{2,1}(Q^{n+1} - Q^n) + \alpha_{2,2}(Q^n - Q^{n-1})}{\Delta t} \qquad (46)$$

32

The flux at t^{n+1} is obtained by using Taylor's expansion about t^n.

$$\vec{f^{n+1}} = \vec{f^n} + (\frac{\delta \vec{f}}{\delta Q})^n (Q^{n+1} - Q^n) + O(\Delta t^2) \tag{47}$$

$$= \vec{f^n} + \hat{A} \Delta Q + O(\Delta t^2) \tag{48}$$

where $\hat{A} = \frac{\delta \vec{f}}{\delta Q}$ is the flux Jacobian matrix. This is split into two to account for the upstream and the downstream signal propagation.

$$\hat{A} \Delta Q = \hat{A}^+ \Delta Q^+ + \hat{A}^- \Delta Q^- \tag{49}$$

Substituting this relation into the governing equation for temporal integration, and rearranging, the result in matrix notation can be expressed as,

$$LHS(Q^{n+1} - Q^n) = RHS \tag{50}$$

Computation of the analytical viscous and inviscid flux Jacobians, due to the nature of the Riemann solver is expensive. Consequently, the Jacobians of the van Leer splitting methods (van Leer (1982)) are used. The splitting involves a temporal damping term which tends to make the LHS diagonally dominant. The flux Jacobian matrix $\hat{A}$ is split as,

$$\hat{A}^{\pm} = \hat{A} \pm I\beta\lambda_{max} \tag{51}$$

where $\lambda_{max} = \vec{V} \cdot \hat{n} + c$ where $\vec{V}$ is the velocity vector, c is the speed of sound, and β is the temporal damping term. The viscous fluxes ensure robustness and are split using a simplified eigenvalue approach. Temporal damping may be used when large time steps are employed.

The resulting matrix is solved iteratively using a symmetric Gauss-Seidel procedure. Newton sub-iterations are used to increase the temporal accuracy of the unsteady problems. This implicit scheme is stable for any Courant-Freidrichs-Lewy (CFL) number, but generally a value of 10^6, is used as recommended by Strang et al. (1999). A block Gauss-Seidel technique is used over a full Gauss-Seidel technique owing to better parallel performance.

Cobalt was rewritten in Fortran 90 to take advantage of various features such as runtime usage of memory and parallelization of the code. The parallel version is based on domain decomposition of the grid, with each processor operating on each zone or subsection of the grid. The code uses Message Passing Interface (MPI) to pass information between processors. The conserved variables of each cell, the initial conditions for Riemann problem for each face and the viscous fluxes are passed between processors through the zone boundaries.

A preprocessing program is used to partition the grid into zones. The intermediate decomposition is performed using ParMETIS Karypis et al. (1997). This algorithm gives almost equally balanced zones with minimal zone interface. The sizes of the zones are almost identical and excellent load balancing is achieved between the processing nodes.

33

Owing to parallel processing the full Gauss Seidal method is observed to be time consuming due to serializing as the number of processors increase. An alternate scheme – block Gauss-Seidal – is used which is more efficient as the information is passed between the zones only after each sweep. This property gives it a Jacobi character at the zonal boundaries.

6 Results and Discussion

6.1 Test conditions

The simulations and analysis reported in this work has been carried out for angles-of-attack of 60° and 90°, at a Reynolds number of 2.1×10^6. Rotary-motion experiments which were carried out by Pauley *et al.* (1995) provide a comprehensive database for the flow around circular and square forebodies at various flow Reynolds numbers and over a range high angles-of-attack. Several other forebody configurations with strakes, trips and the effect of the aftbody were also studied by the experimentalists, thus providing a thorough characterization of the surface pressure, force, and moment coefficients.

The predictions of the flow are obtained for fully turbulent boundary layers, initiated by specifying at the inlet a small level of eddy viscosity. A non-dimensional time step of 0.025 is used for the calculations. The pressure coefficients are based on the aerodynamic pressure $0.5\rho_\infty U_\infty^2$ and the force coefficients based on the area Db. Spin coefficient have been calculated as $\Omega b/(2U_\infty)$, Ω being the rotation rate. Pressures were measured in the wind tunnel tests at eight stations, located at $x/L = 0.027, 0.055, 0.111, 0.167, 0.222, 0.305, 0.805$ and 0.903. Measurements of the yawing moments and the side forces were also reported. The simulated results have been compared with the reported pressures and forces.

6.2 Grids

Unstructured grids was generated using Gridtool and VGRIDns following the procedures detailed previously. A cubic domain with dimensions equal to 20 times the ogive length were used. The co-ordinate system was defined such that the origin lies at the center of the body. The axis of the body coincided with the x-axis and the y-axis represents the streamwise direction (i.e., aligned with the freestream velocity vector).

A baseline grid (Figure 20) with approximately 6.5×10^6 cells was initially generated. The grid consists of prisms near the surface, tetrahedron in the outer domain, and pyramids in the zone between the prisms and tetrahedron. Half the geometry is initially gridded and later mirrored in Blacksmith. The wall-normal spacing to the cell nearest the wall around $y^+ = 1$ is provided from the surface, and the grid is stretched at a rate of 1.2 in the boundary layers. These values were consistent with the DES grid specifications prescribed by Spalart (2001). A set of 15-20

prism layers are generated by VGRIDns and the rest of the domain is meshed by tetrahedron using the advancing front algorithm. The grids in between the tetrahedron and the prism layers were combined to give more isotropic pyramid cells. Grid cells are also clustered in the vicinity of the nose in order to accurately resolve flow variations in this region.

A coarser grid with around 2.1×10^6 cells was created by increasing the size of the sources. A dense grid was created by changing the value of 'ifact' to 0.9 in the d3m file. The dense grid comprised of 8.75×10^6 cells as shown in Figure 21. The denser grid was not an adapted version of the baseline grid, but an overall refinement of the baseline grid. Calculations were carried out on these grids to assess the sensitivity of the simulation strategy to grid refinement. The value of C_{DES} is set to 0.65 as recommended by Shur *et al.* (1999).

6.3 Numerical specifications

Apart from the grid the other user-defined parameters in it Cobalt are the timestep, number of matrix sweeps, and the number of sub-iterations. Forsythe (2000) explored the effect of the number of sweeps and observed that increasing the number of iterations beyond 32 yielded no further improvement in the computations. Forsythe (2000) also reported that since every sub-iteration recomputes the Jacobian matrix, that process carries approximately the same cost as a single iteration. For the present work the number of matrix sweeps has been fixed at 24 and the number of sub-iterations set to 3 for all of the runs. For the rotating cases the number of sub-iterations is increased to 5, a value arrived at following numerical experiments for geometries undergoing prescribed motion. The time step has been fixed at 0.025, non-dimensionalized using the freestream velocity and the width of the forebody. Time averages were acquired for over 100 time units.

6.4 Code specifications

Cobalt allows a variety of boundary conditions. For the current geometry, the farfield has been specified using a modified Riemann invariant, the surface of the body as an adiabatic no-slip solid wall. Flow enters and leaves the domain along the patches defined using the farfield condition. The modified Riemann invariant condition fixes the Riemann invariants for the subsonic inlet case and fixes the pressure at the outlet boundaries at the user specified values, allowing other variables to vary. For a supersonic condition it acts as a simple supersonic entrance/exit condition allowing all variables to vary. An adiabatic no-slip boundary condition at the wall specifies a zero velocity at the wall and assumes a zero normal pressure and density gradient. Computations were carried out in parallel, typically using 128 processors of an IBM SP3. Processing requirements for the baseline grid are 45 seconds per iteration (6.97 μsec/cell/iteration).

6.5 Results

6.5.1 Stationary geomety at $\alpha = 60°$

Predictions of flow around a rotating forebody at 60° angle of attack using the Baldwin-Lomax turbulence model van Dam *et al.* (2001) showed encouraging results. This prompted the use of S-A RANS model to simulate the flow around a stationary forebody at 60° angle of attack.

The flow is characterized by attached boundary layers around each of the four corners of the forebody, while separation is observed around the aftbody. Coherent structures in the leeward side are resolved which result in a pair of symmetric counter-rotating vortices in the wake, as shown by the eddy viscosity contours in Figure 22. The surface pressure also shows signatures of these structures, displaying striations corresponding to pressure variations indicating the presence of some coherent structures over the forebody and a nearly constant pressure along the later half of the body consistent with massive separation.

Pressure distributions

Pressure distributions were measured at eight axial stations (Figure 22) and compared with the experimental measurements. The angle (θ) around the forebody have been measured in the clockwise direction with reference to the windward symmetry plane.

The URANS predictions are compared with the experimental data at the axial stations in Figures 23-26). For the six stations on the forebody the flow remains attached from the stagnation point. The pressure achieves a minima (suction) as it negotiates the corner, at $\theta = 50° - 54°$ and $\theta = 303° - 307°$. The pressure coefficient values at the corners vary from -2.4 at station 1, -2.25 at station 2, and around -2.0 for the other stations on the forebody. The pressure increases as the flow moves up the vertical (aligned with the freestream velocity) face of the forebody and reaches a peak at the center. The signature of counter-rotating vortices are apparent on the leeward side. This suction is comparable in magnitude to the suction on the windward side for the first three stations of the forebody. The secondary suction C_p increases from a value of -2.44 at station 1 to a value of -1.45 at station 6. The flow in the wake of the forebody is characterized by significant variations in the surface pressure, which are the signatures of the coherent structures that develop in the aft region.

Progressing from the nose of the forebody towards the center, the flow close to the surface loses sufficient momentum that it separates as it reaches the aftbody. This momentum loss is evident from the increase in the pressures at the leeward corners along the forebody stations and ultimately a region of separation (constant C_p) at stations 7 and 8. At these two stations where measurements were acquired on the aftbody, the secondary suction is absent. The suction is observed at around $\theta = 54°$ and 303° for both the stations. The C_p values are observed to be about -1.84 at $x/L = 0.805$ and -1.6 at $x/L = 0.903$. Separation is observed between $\theta = 126° - 232°$ in both the stations. Local suction is observed at $\theta = 120°$ and 240°, as the flow turns the corners in the leeward side of

the forebody.

Overall, the URANS predictions are in good agreement with the experimental measurements. Boundary layer prediction by the RANS model is adequate and the coherent structures characterizing the wake structure can be reasonably represented.

6.5.2 Stationary geometry at $\alpha = 90°$

Role of the model – URANS and DES

The success of the URANS model for the earlier case does not guarantee accuracy in a blind application of this model at higher angle of attack, due to the effects of massive separation. S-A URANS and DES predictions were obtained for the baseline grid to test the capabilities of the models, in predicting the flow around a stationary forebody at 90° angle of attack.

The contours of the eddy viscosity ratios in planes corresponding to the eight stations where pressures are measured (Figure 27) depict the different flow structures predicted by the RANS and the DES models. The surface of the forebody in the figure is colored by pressure. On the leeward side of the forebody the URANS model predicts a pair of large counter-rotating vortices, while the flow predicted using DES is more chaotic and exhibits a more pronounced three-dimensional character. The surface pressures also show the signature of the flow structure in the wake. The URANS predictions resemble those shown above at lower angle of attack ($\alpha = 60°$), with relatively strong variations in the pressure in the aft region showing the inability of the URANS approach to reliably model for massively separated flows. The DES prediction shows a relatively uniform pressure on the lee sides.

Pathlines predicted by the simulation techniques are shown in Figure 28 and again reinforce the large differences in the flow structure predicted using the two techniques. DES predicts a low pressure region in the forebody wake, which results in a flow from the aftbody towards the forebody. The URANS case predicts a low pressure region in the center of the body resulting in an axial flow from both the forebody and the endcap. This difference in the flow structure leads to different axial forces predicted using the two models as summarized in Table 1 and Figure 29].

Force/Moment Histories

The time histories of forces and yawing moments predicted using DES and URANS are shown in Figure 29. The variation of the yawing moment and the side force exhibit similar behavior but the variation exhibited by the URANS prediction is substantially less pronounced than the DES results. The flowfield predicted by the URANS model is periodic, while the DES exhibits more chaos. The root-mean-square values of the yawing moment and the side forces in Table 2 shows that the variations in the DES predictions are substantially larger than the URANS results. The streamwise force (drag) predicted by the URANS case of 0.334 does not drastically differ from the DES prediction of 0.321. The largest difference is observed in the axial forces computed by

37

Case	axial force	streamwise force
DES coarse	0.0830	0.3244
DES baseline	0.0857	0.3212
DES fine	0.0853	0.3217
URANS baseline	0.022	0.334

Table 1: Mean axial and drag force.

Case	side force (rms)	yawing moment (rms)
DES coarse	0.031	0.016
DES baseline	0.035	0.020
DES fine	0.037	0.019
URANS baseline	0.0068	0.0043

Table 2: Root-mean-square side force and yawing moment.

the two approaches. The axial force calculated by DES is four times the URANS prediction owing to the difference in the flow structures (c.f., Table 1, Figure 28). The surface pressures predicted by DES on the leeward side along the forebody are higher compared to the URANS calculations, resulting in a higher axial force in the DES case.

Pressure distributions

The comparisons of the DES and the URANS predictions with the measured values for the stationary (non-rotating) geometry are shown in Figure 30-Figure 33. The computations and the measurements show that the flow from the stagnation point, $\theta = 0°$, along the forebody sections is attached. A minima in the pressure coefficient is observed in the vicinity of $\theta = 48° - 56°$ and symmetrically on the other side at $\theta = 303° - 310°$. This region corresponds to the smooth windward corners of the forebody where the flow tends to accelerate. The magnitudes of the minima range from around -2.25 to -2.50.

Another pair of suction minima are observed in the C_p distributions as the flow negotiates the corners on the leeward side at angles $\theta = 118° - 120°$ and $\theta = 240° - 242°$. These secondary minima become more pronounced from station 1 ($x/L = 0.027$) to station 8 ($x/L = 0.903$), indicating the presence of a secondary (axial) flow. The suction in the leeward corners at the stations on the aftbody ($x/L = 0.805$ and 0.903) are comparable in magnitude to the suction in the windward side, due to the presence of the endcap. The variation in the secondary suction (in the leeward side of the body) along the forebody and the endcap shows the effect of the flow from the endcap. From the pressure profiles separation is predicted at $\theta = 127° - 132°$ and $229° - 232°$

38

on the forebody. Compared to the axial stations along the forebody the flow around the aftbody is complex and yields more variation with θ.

DES predictions of the pressures are generally in good agreement with the experimental measurements. A fully turbulent boundary layer is established around the forebody, which remains attached along the windward side. URANS predictions exhibit significant variations in the pressures around the forebody owing to the coherent structures in the leeward region. The pressures are slightly over-predicted in the aftbody at station 7 ($x/L = 0.805$) at angles 90° and 270°. At the last station, the DES prediction seems to be in better agreement with the measurements as compared to the URANS case.

Skin friction coefficients

The skin friction distributions (Figures 34-37) exhibit similar characteristics as the pressure plots. It is observed that, for the DES predictions along the forebody, the skin friction increases as the flow negotiates the corner of the forebody. The flow decelerates along the sides of the body and the boundary layer thickens leading to an abrupt drop in the C_f values. As the flow turns the leeward corners and the boundary layer separates, the skin friction is negligible. Similarly, the RANS predictions shows coherence in the wake, though in most of the stations the predictions in the windward side are consistent with the DES results. The C_f plot at station 1 (Figure 34) also shows the earlier separation of the boundary layer which might have been due to the proximity the RANS-LES interface to the solid walls near the nose. Along the aftbody, both the DES and the URANS predictions are similar, and the C_f values show attached flow in the leeward side of the aftbody.

RANS-LES interface

Because DES is not a zonal technique, the RANS and the LES regions are demarcated based on the grid spacing. The natural applications of DES requires the boundary layer to be calculated in the RANS mode while the predictions in the regions away from the wall should be predicted in the LES mode of the technique. This heightens the importance of grid generation and the burden on the user to generate appropriate meshes.

Figure 38 shows the RANS-LES interface near the leeward corner, the figure generated using the baseline grid. The boundary layer separates in the vicinity of this region and the RANS-LES interface lies inside the boundary layer. Figure 39 shows the variation of the turbulent length scale and the eddy viscosity across the interface. The increase in the length scale (d) is initially sharp (i.e., the wall distance) inside the RANS region which results in the drop of the destruction term of the eddy viscosity transport equation. The comparison of the eddy viscosities in the region close to separation shows the difference in the eddy viscosities predicted for the DES and the RANS cases. The continuous increase in the eddy viscosity in the RANS prediction diffuses the eddies away from the wall, while in the DES, the eddy viscosity decreases as the length scale is redefined

39

in the LES region, allowing the eddies to develop.

The effect of grid refinement on the DES predictions is shown in Figures 41-42 where the instantaneous vorticity magnitude is shown in the center plane and in the wake at a distance of $D/2$ from the body for the coarse, baseline, and the fine grid. The figures show that increasing the grid density resolves a wider range of scales. Finer grids in the wake, where the LES mode is active, yields an extended range as compared to the coarser-grid runs. The baseline grid seems to predict an adequate range of scales. The plane in the wake also emphasizes the resolution of more eddies with grid improvement. A plot (Figure 40) of the pressures calculated at axial station 1 ($x/L = 0.027$) for the coarse, baseline, and the fine grids show the convergence of the results to the experimental values, with the increase in grid density.

6.5.3 Rotary motion at $\alpha = 90°$

A challenging case – forebody at 90° angle of attack and undergoing prescribed rotary motion at a spin coefficient of 0.2 ($\frac{\Omega L}{2U_\infty} = 0.2$), was considered to test the performance of DES for a configuration relevant to aircraft spin.

The contours of the instantaneous vorticity magnitude, for the baseline grid, at three planes are shown in Figure 43. The contours plotted at $x/L = 0.222$ (on the forebody section), $x/L = 0.5$ and 0.805 (on the aftbody) show the influence of the rotary motion on the vorticity shed in the wake. The flow shows skewing towards opposite sides on the front and the rear stations, as dictated by the local conditions at that instant. The windward-side suction peak is greater than occurring on the leeward side, resulting in yawing moments larger in magnitude as compared to the stationary case.

Figure 45 shows the DES predictions of the pressure coefficient at $x/L = 0.111$. The pressure distribution is no longer symmetric due to the effect of rotation. DES predicts a larger separation compared to the experiments, resulting in a mismatch at $180° < \theta < 270°$. The C_p minima occurs at around $\theta = 306°$ in the vicinity of the rear windward corner of the forebody. The value calculated by DES is -2.6 against the experimental value of -2.9. The C_p value at the stagnation point is 0.61.

For the axial station at $x/L = 0.166$ (Figure 45) the DES predictions capture the behavior observed in the experiments. Minima in the C_p values around both the windward corners is recovered. The separation predicted also improved as compared to the calculations at $x/L = 0.111$, and is in good agreement with the measured pressure distribution. The distributions indicate an attached flow around the windward corners with separation in the leeward side. The uniform distribution calculated along the leeward side in the separated region is also accurate with $C_p = -0.38$. At the stagnation point $C_p = 0.81$ which is in good agreement with the experimental value.

The pressure distribution along the last two axial stations on the forebody (Figure 46) also show good agreement with the measured results. DES accurately predicts the secondary minima which

40

is induced by the interaction of the boundary layer with the leeward corners of the body. The C_p values for the stagnation point are 0.92 and 1.0 respectively for the last two stations. The suction minima for all the stations lies at around $\theta = 304° - 310°$, at almost the points where the suction minima was also noted in the stationary case.

7 Conclusions

An analysis of the flow around stationary and rotating forebodies has been carried out using DES and URANS. The comparison of the qualitative and quantitative results show very agreement with the experimental observations for the stationary geometries at both 60° and 90° angle of attack. For the configuration undergoing prescribed rotary motion, the agreement with measurements is less satisfactory though the main features of the effect of rotation are recovered in the computations.

While the flow at 60° angle of attack can be predicted to useful accuracy using URANS, the flow at $\alpha = 90°$ experiences massive separation, favoring the application of DES over URANS. In cases of massive separation, the region of interest is the separated wake, which forms the LES focus region. In such cases, the loss of information due to averaging in the RANS region is not significant. The redefining of the length scale for the DES cases, draws down the eddy viscosity in the regions away from the wall, allowing the eddies to develop in these regions.

Accurate DES prediction of massively separated flows requires the appropriate positioning of the RANS-LES interface. The definition of the interface is fixed by the maximum inter-centroidal distance, typically the streamwise or the spanwise grid spacings, and these spacings should be coarse enough in the wall region such that the RANS region comprises most of the boundary layer. If the interface is close to the wall, the eddy viscosity is decreased by the action of the DES limiter, possibly resulting in early separation of the boundary layer. For the present computations, the eddy viscosity and the turbulent length scale (Figure 39) shows the drop in the eddy viscosity in the LES region. By comparison, the RANS eddy viscosity increases away from the wall. The trend in the plots for the DES and the RANS cases seem to be similar until the interface is reached, though the deviation begins in the vicinity of the interface. Locating the interface close to the boundary layer edge takes advantage of the efficiency of RANS models in these regions.

DES and URANS were applied to prediction of the massively separated flow around an ogive characterized by a rounded cross section. Angles of attack of 60° and 90° were considered and for the highest Reynolds number at which measurements were acquired, $Re = 2.1 \times 10^6$. DES predictions of the pressure distribution for the static-geometry flow and ogive undergoing rotary motion are for the most part in relatively good agreement with measurements, improved agreement noted for the static-geometry flow. For $\alpha = 60°$, the flow structure in the wake was characterized by a pair of counter-rotating vortices that strongly influence the leeward pressure distribution. The relatively coherent wake for $\alpha = 60°$ resulted in URANS predictions of the pressure distribution

on the forebody that were essentially the same as the DES results.

For the higher angle of attack, $\alpha = 90°$, and the flow around the static geometry, DES predictions are far superior to the result obtained using unsteady RANS. URANS predictions of the flow in the wake region are overly coherent, the wake structure in the URANS again characterized by a pair of counter-rotating vortices as obtained in the RANS at $\alpha = 60°$. Strong three-dimensionality along the forebody is not recovered in the URANS, leading to a pressure distribution that has substantial variation in the separated region, rather than the uniform distribution measured in the experiments reported by Pauley *et al.* Pauley *et al.* (1995) and predicted in the DES.

The ogive undergoing rotary motion was computed using an ALE formulation, applied to the ogive for the present investigations for rotation about the model center at spin coefficients of 0.1 and 0.2. Visualizations of the vorticity magnitude in the leeward region showed the skewing of the wake by the rotation. Pressure distributions on the forebody exhibit adequate agreement with measured values, the asymmetry induced by the rotation being recovered and the overall variation with θ being captured in the DES, though the predictions were less satisfactory as compared to the static-geometry results.

DES predictions of the static geometry were obtained for a range of mesh resolutions, the calculations showing that a deeper structure is resolved in the wake. This is an important attribute for hybrid RANS/LES methods, demonstrating that in the limit of very fine grids the role of the turbulence model would vanish and the technique approaches Direct Numerical Simulation. In general, the three-dimensionality of the wake was substantially stronger in the DES as compared to the RANS, consistent with related studies Travin*et al.* (2000). Though the wake structure did not exhibit as much axial (spanwise) variation in the URANS results, three-dimensionality was present, an aspect that probably contributes to the relative agreement of the integrated and averaged streamwise force between the DES and URANS. Strong differences in the time-dependent characteristics of the solutions were noted in the rms forces and moments. An additional and important difference between the DES and URANS noted in the current study was the pressure distribution along the forebody. This contributed not only to large differences in the axial force but would also influence quantities such as the pitching moment. These features are problematic for usage of URANS in flight applications focusing on areas such as control and stability, such applications should substantially benefit from the higher fidelity offered by DES.

The present computations were performed of the flow with fully turbulent boundary layers, accomplished by seeding the inflow condition with a small level of eddy viscosity, sufficient to activate the turbulence model as the fluid entered the boundary layer. Measurements at lower Reynolds numbers showed strong *Re* effects Pauley *et al.* (1995) and such regimes comprise an important and challenging test case for hybrid methods. Effects of transition to turbulence, possibly inter-mingled with boundary layer separation, are exceedingly difficult to model and accuracy requirements are typically very high. The experimental database will prove useful for simulation

efforts that consider the lower range in Reynolds number. DES (and S-A RANS) calculations of the lower Reynolds number range performed using the tripless approach developed by Travin *et al.* Travin*et al.* (2000) will be very useful in assessing a simple (but state of the art) approach that accounts for laminar separation with transition in the separating shear layers.

Finally, aircraft forebodies are often asymmetric due to imperfections in the geometry, an effect that can produce a large yawing moment, even for configurations without sideslip. The strong yawing moment on low aspect ratio aircraft such as the F-15E can lead to relatively flat spins, for example. The computational methodologies under development and assessment in this work, while not fully complete, will be important for accurately modeling such phenomena.

References

Spalart, P. R. , Jou W-H. , Strelets M. , and Allmaras, S. R., "Comments on the Feasibility of LES for Wings, and on a Hybrid RANS/LES Approach," *Advances in DNS/LES, 1st AFOSR Int. Conf. on DNS/LES*, Aug 4-8, 1997, Greyden Press, Columbus Oh.

Spalart, P.R., and Allmaras, S.R., " A One- Equation Turbulence model for Aerodynamic Flows." *La Recherche Aerospatiale*, No 1, 1994, pp 5-21.

Fisher, D.F., Richwine, D.M., Landers, S., "Correlation of Forebody pressures and aircraft yawing moments on the X-29A Aircraft at high angles of attack", *NASA Technical Memorandum 4417*

Bjarke, L.J., Del Frate., J.H., Fisher, D.F., "A summary of the forebody high angle-of-attack aerodynamics research on the F-18 and the X-29A aircraft", *NASA Technical Memorandum 104261*

Forsythe,J.R., Squires,K.D., Wurtzler,K.E., and Spalart,P.R., "Detached-Eddy Simulation of fighter aircraft at high alpha", *AIAA Paper 2002-0591*, 2002.

Squires K.D., Forsythe, J.R., and Spalart,P.R., "Detached-Eddy Simulation of the separated flow around a forebody cross-section", *Large-Eddy Simulation IV, ERCOFTAC Series – Volume 8*, B.J. Geurts, R. Friedrich and O. Metais, editors, Kluwer Academic Press, pp.481-500, 2001.

Pauley, H., Ralston, J. and Dickes, E., "Experimental Study of the Effects of Reynolds number on high Angle of Attack Aerodynamic Characteristics of Forebodies during rotary motion", *NASA CR 195033*, Jan. 1995.

van Dam, C.P., Saephan, S., Fremaux, C.M., DalBello, T., "Prediction of Flows about Forebodies at high Angle of attack Dynamics Conditions". Presented at *NATO/RTO Applied Vehicle Technology Panel Symposium on Advanced Flow Management: Vortex Flow and High Angle of Attack*, May 7-10, 2001, Leon, Norway. Paper No: MP-96-P39.

Forsythe, J.R., "Numerical Computation of Turbulent Separated Supersonic Flowfields", *PhD Thesis, Wichita State University*, Spring 2000

Pirzadeh, Shahyar., "Structured background grids for generation of unstructured grids by advancing front method", AIAA Journal, Vol 31, No 2, Febraury 1993, pp 257-265.

Pirzadeh, Shahyar., "Unstructured Viscous Grid Generation by Advancing-Front Method", NASA CR 191449, Apr 1993

Spalart, Philippe.R., "Young-Person's Guide to Detached Eddy Simulation Grids", *NASA/CR-2001-211032*, July 2001.

Strang, W. Z., Tomaro, R. F, Grismer, M. J., "The Defining Methods of Cobalt$_{60}$: a Parallel, Implicit, Unstructured Euler/Navier-Stokes Flow Solver," *AIAA 99-0786*, January 1999.

Samareh-Abolhassani, Jamshid., "GridTool: A Structured Modeling and Grid Generation Tool", *Proceedings of the Workshop on Surface Modelling. Grid Generation, and Related Issues in CFD Solutions, NASA CP -3291*, May 1995

Strang,W.Z., "Parallel Cobalt User's manual", June 1999.

Garriz, Javier A., "VGRID 3.2 - Reference Documents", Sep 1998.

Samareh-Abolhassani, Jamshid., Unstructured Grid on NURBS surface, *AIAA paper 93-3454.*

Pirzadeh, S., "Vortical flow prediction using an adaptive unstructured grid method", *Research and Technology Organization Appied Technology Panel Meeting*, Norway, 7-11 May 2001.

Peraire, J., Peiro, J., Morgan, K., "Advancing Front Grid Generation", *Handbook of Grid Generation*, CRC Press, pp. 17-1 - 17-22.

Hoffman, K.A., Chiang, S.T., Computational Fluid Dynamics - Volume II, Engineering Education System, Third Edition, 1998, pp 312 - 319.

Spalart, Phillipe., "Strategies of turbulence modelling and simulations" *Int. J. of Heat and Fluid Flow 21*, 2000.

Squires,K.D., Forsythe,J.R., Morton,S.A., Strang,W.Z., Wurtzler,K.E., Tomaro,R.F., Grismer,M.J., and Spalart,P.R., "Progress on Detached-Eddy Simulation of massively separated flows",*AIAA Paper 2002-1021*, 2002.

Morton,S.A., Forsythe,J.R., Squires,K.D., and Wurtzler,K.E., "Assessment of unstructured grids for Detached-Eddy Simulation of high Reynolds number separated flows", *Proceedings of the Eighth International Conference on Numerical Grid Generation in Computational Field Simulations*, 2002.

Squires,K.D., Forsythe,J.R., Morton,S.A., Blake,D.C., Serrano,M., Wurtzler,K.E., Strang,W.Z., Tomaro,R.F., and Spalart,P.R.,"Analysis of full aircraft with massive separation using Detached-Eddy Simulation", *Proceedings of the High Performance Computing Modernization Program 2002 Users Group Conference*, Austin, Texas, 2002.

Shur, M., Spalart, P. R., Strelets, M., and Travin, A, "Detached-Eddy Simulation of an Airfoil at High Angle of Attack", *4th Int. Symp. Eng. Turb. Modelling and Measurements*, Corsica, May 24-26, 1999.

Travin, A., Shur,M., Strelets,M., Spalart,P.R., "Detached Eddy Simulations past a Circular Cylinder", *Flow, Turb and Combustion* 63 2000.

Spalart., Philippe R., Boeing Corp., Seattle, "Detached-Eddy Simulation", Invited Talk, *IAM-PIMS Joint Distinguished Colloquium Series*, Oct 2001.

Hirsch, C., "Numerical computation of internal and external flows - Vol 2 Computational Methods for Inviscid and Viscous flows.", John Willey and Sons New York, 1990.

Colella, Phillip., "Glimm's method for gas dynamics", *SIAM Journal of Stat. Comput.* March 1982.

Godunov, S.K., "A difference scheme for numerical computation of discontinuous solution of hydrodynamic equations", *Sbornik Mathematics*, 47:271-306, 1959.

J.J. Gottlieb and C.P.T. Groth., "Assessment of Riemann solvers for unsteady one-dimensional inviscid flows of perfect gases." *Journal of Computational Physics*, 78:437-458, 1988

Bram van Leer., "Flux-vector splitting for the Euler equations", *Eighth International Conference on Numerical Methods in Fluid Dynamics.* Aachen, Germany, June 28 - July 2, 1982.

Barth, T.J., "Numerical Aspects of computing viscous high Reynolds number flows on unstructured meshes", *AIAA-91-0721*

Glimm,J., Commun. Pure Appl. Math, 18, 697 (1965).

Karypis, G., Schloegel, K., and Kumar, V., *ParMETIS: Parallel Graph Partitioning and Sparse Matrix Ordering Library Version 1.0.* University of Minnesota, Department of Computer Science, Minneapolis, MN 55455, July 1997.

Wurtzler.K.E., "An effectiveness study of F-15 forebody flow analysis using $Cobalt_{60}$, *AIAA 99-0536*
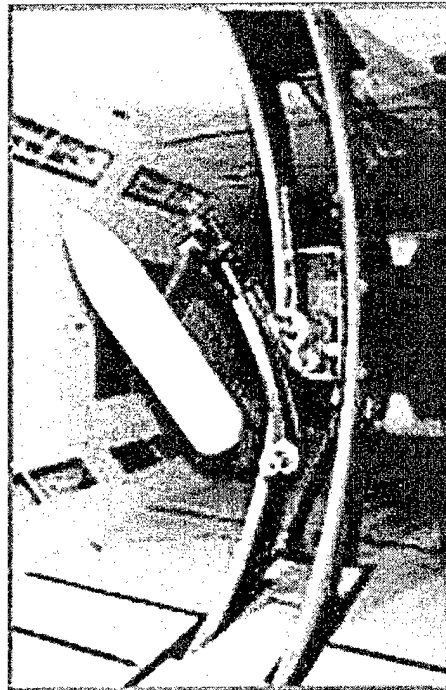
Figure 1: Experimental setup at DeRA, United Kingdom. (from www.bihrle.com/services_wtc1.html).
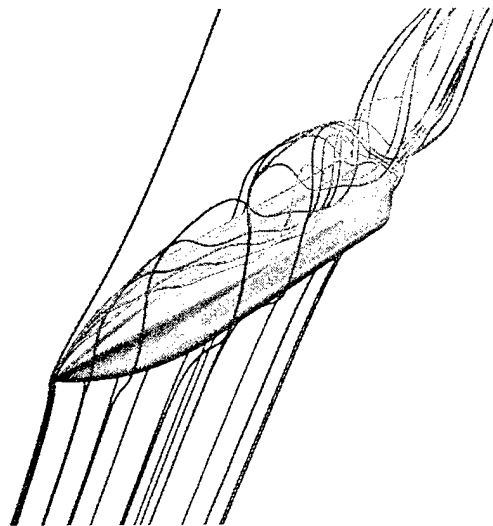
Figure 2: Flow around a forebody at 60° angle-of-attack.
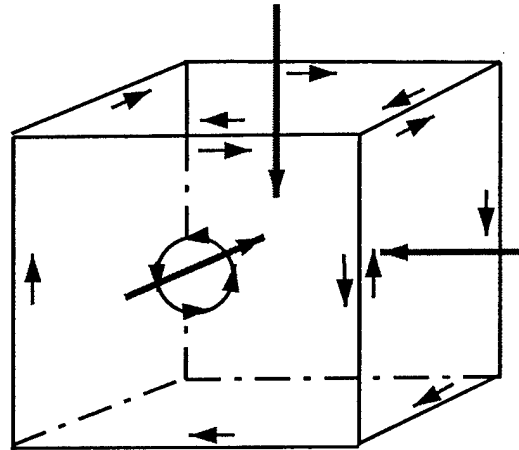
Figure 3: Orientation of patches and the direction of grid growth.

(a) (b)
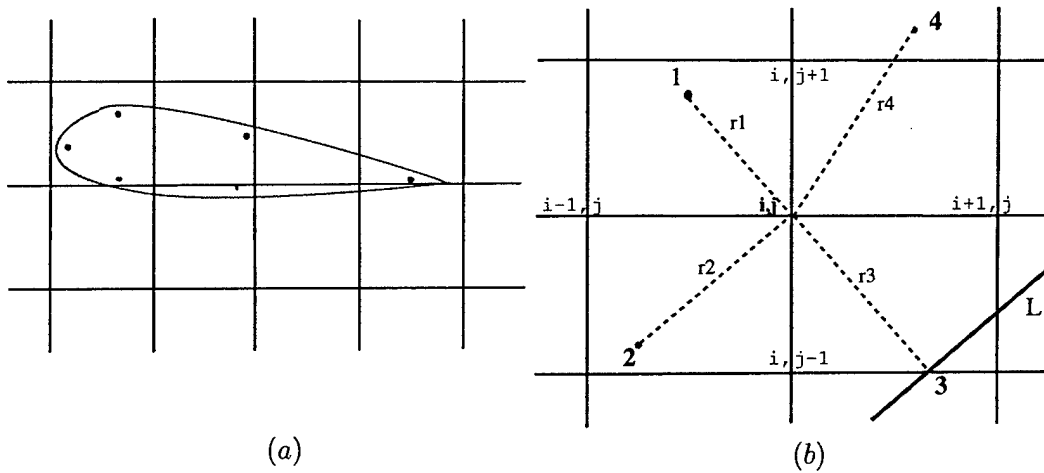
Figure 4: (a) Airfoil with sources on a Cartesian background grid; (b) Effect of sources on a Cartesian grid point $G_{i,j}$.
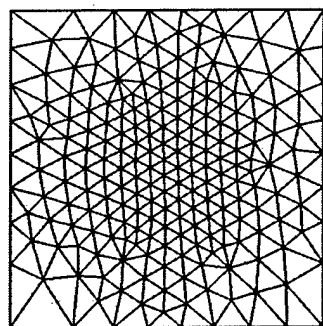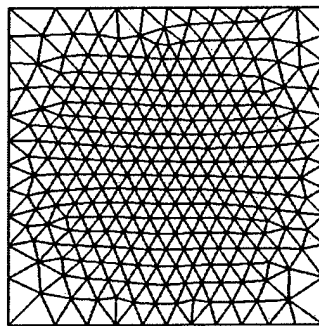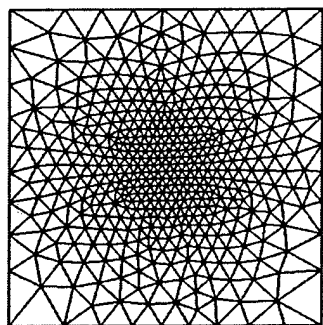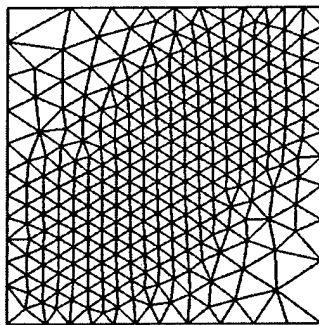
Figure 5: Effect of source sizes(s) and strengths (an, bn). (a) an = 1, s = 1 (b) an = 2.5, s = 1 (c) an = 1, s = 0.5 (d) an = 1, s = 1, bn = 1 (directional bias).
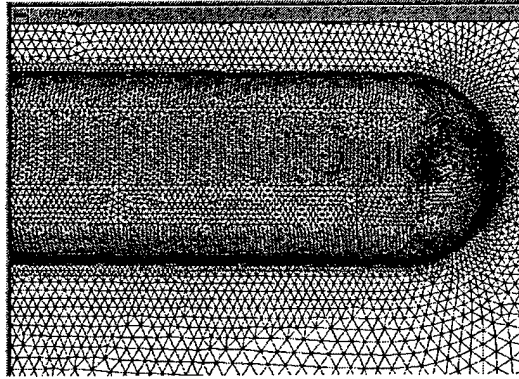
Figure 6: Surface grids in VGRIDns.

(a)   (b)   (c)
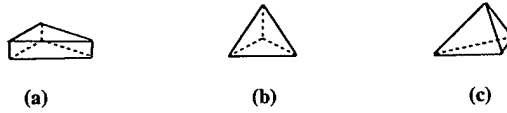
Figure 7: Types of cells used (a) prisms; (b) pyramids; (c) tetrahedron.
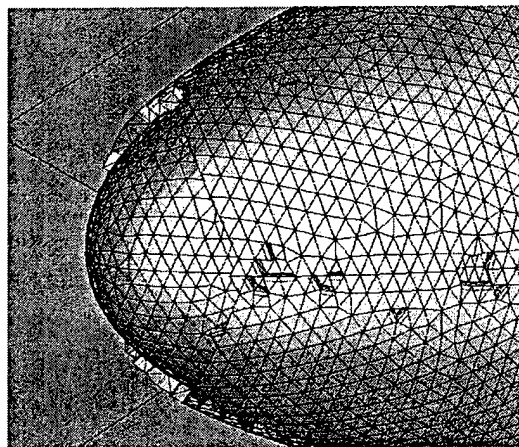
Figure 8: Prism layers.
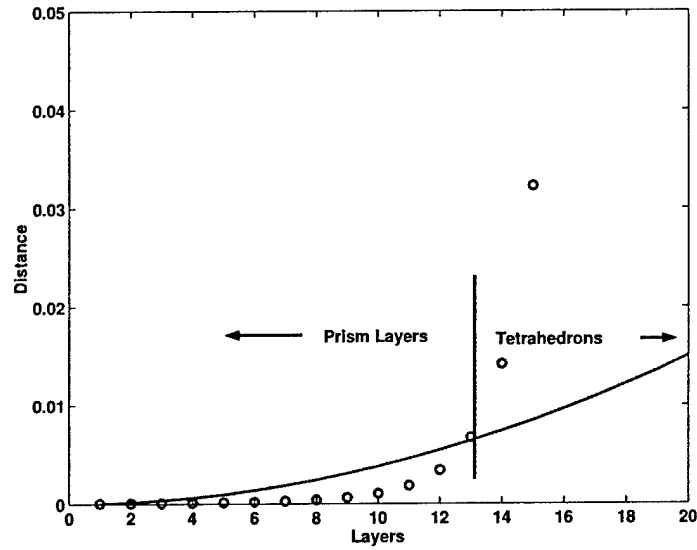
Figure 9: Prism layers generated until the size exceeds the size determined by the sources. ————- Size determined by source, o - Size of prism layers.
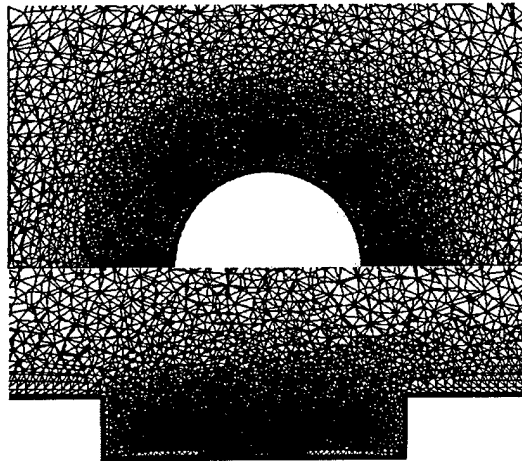
Figure 10: Grid in an axisymmetric cavity adaptively refined based on vorticity – lateral and cross-sectional views.
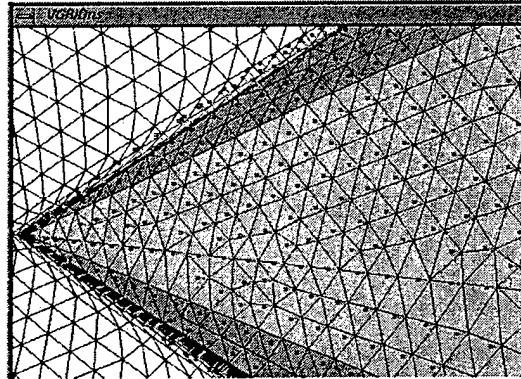
Figure 11: VGRIDns – surface vectors on the geometry.
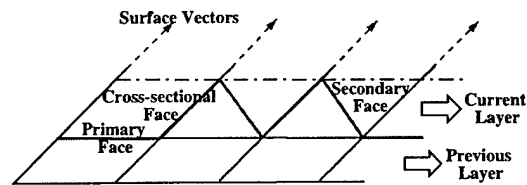
Figure 12: Advancing layers.
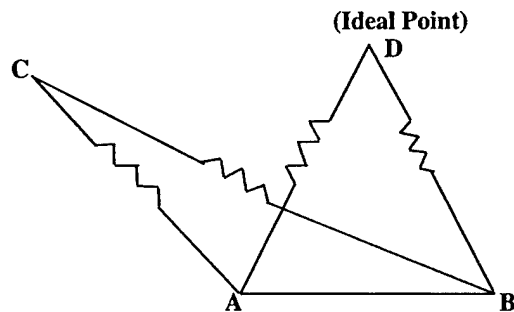
Figure 13: Spring analogy.

51
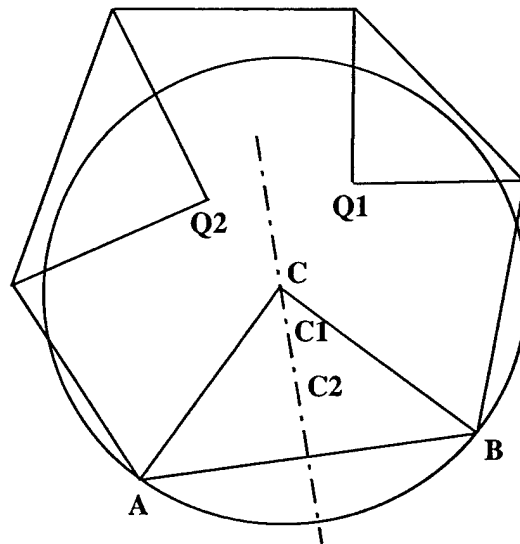
Figure 14: Advancing Front – selection of grid point.
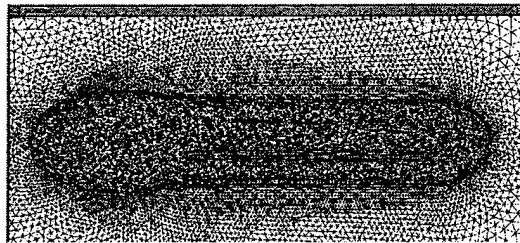
Figure 15: VGRIDns - Advancing Fronts around an ogive.
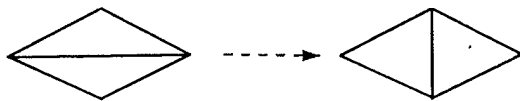
Figure 16: Swapping of diagonal to attenuate distortion.
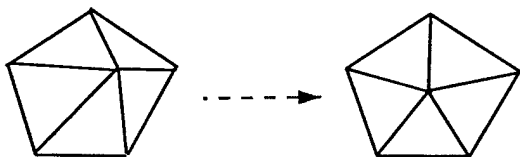
Figure 17: Smoothing the mesh to improve grid quality.
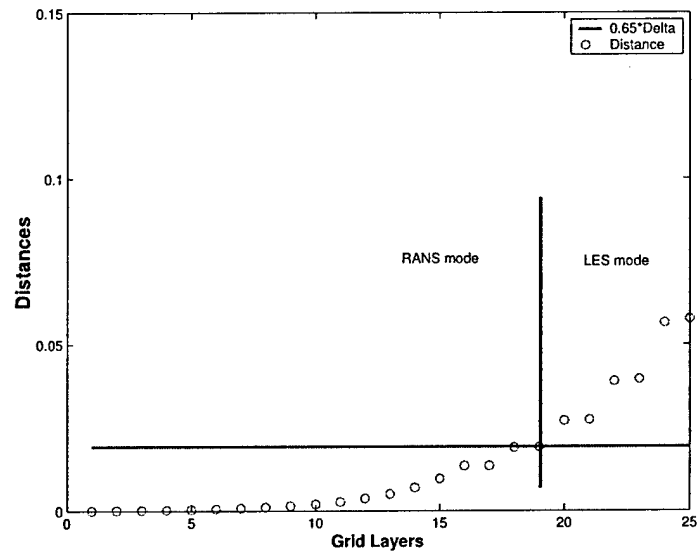
Figure 18: Location of RANS-LES interface.
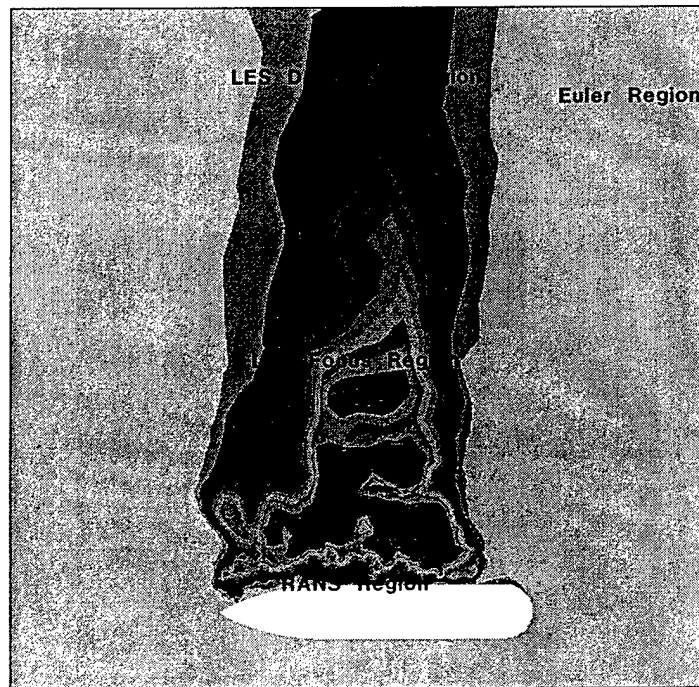
Figure 19: Classification of regions in DES grids.

53
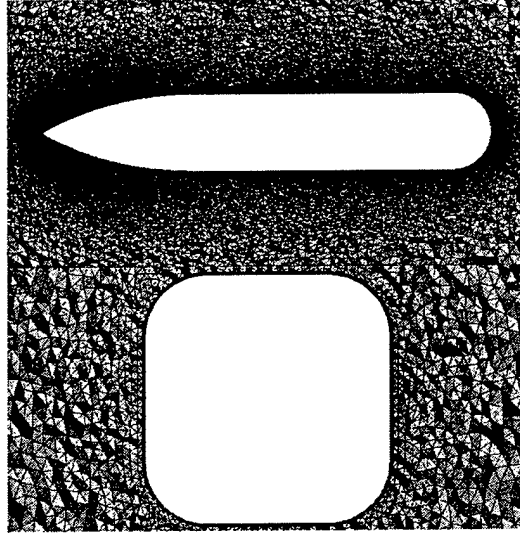
Figure 20: Side and end views of the baseline grid (6.5×10^6 cells). Length of the forebody ogive = $2D$, total length = $6D$. Cross-section is a square with rounded corners, radius = $D/4$.
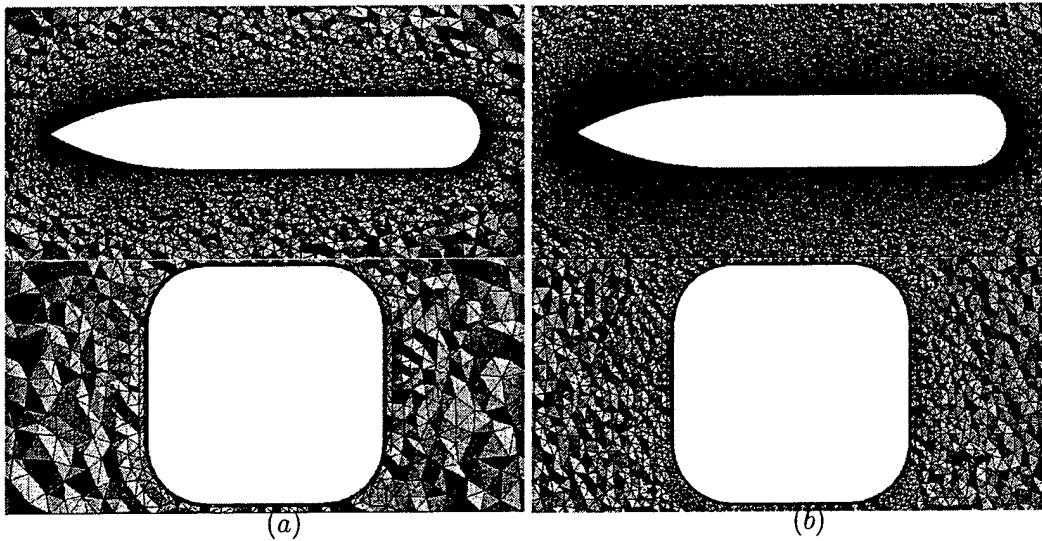
Figure 21: Side and end views of the coarse and fine grids. Coarse grid: 2.1×10^6 cells, Fine grid: 8.75×10^6 cells.
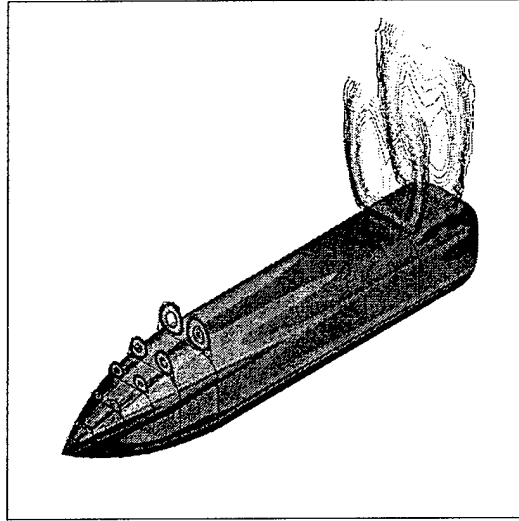
Figure 22: Ratio of the instantaneous eddy viscosity to the molecular value at the eight axial locations for which pressure measurements are available. Geometry colored by pressure.

$$(a) \qquad\qquad (b)$$
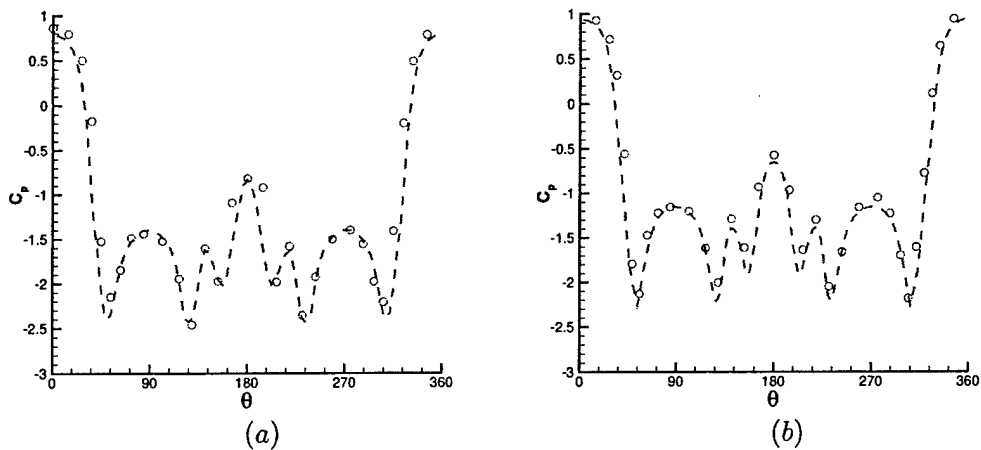
Figure 23: Pressure coefficient at (a) $x/L = 0.027$ and (b) $x/L = 0.055$. o measurements; ---- URANS.

Figure 24: Pressure coefficient at (a) $x/L = 0.111$ and (b) $x/L = 0.166$. o measurements;
---- URANS.

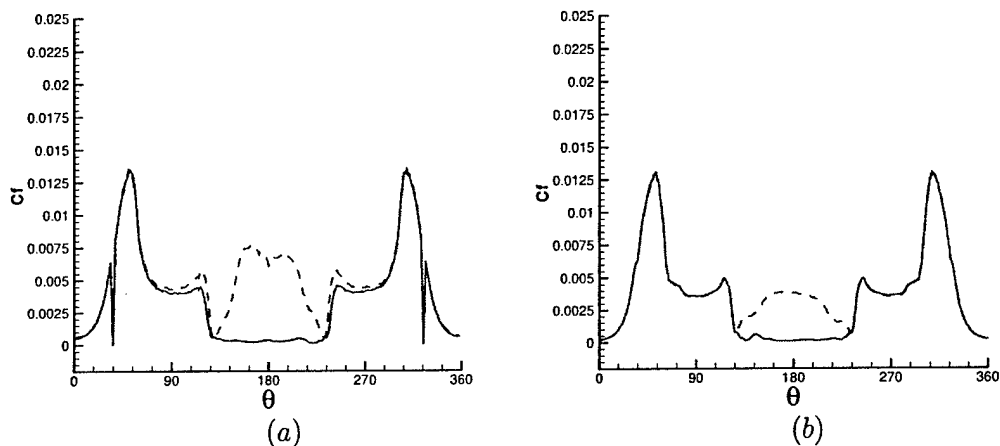Figure 25: Pressure coefficient at (a) $x/L = 0.222$ and (b) $x/L = 0.305$. o measurements;
---- URANS.

56
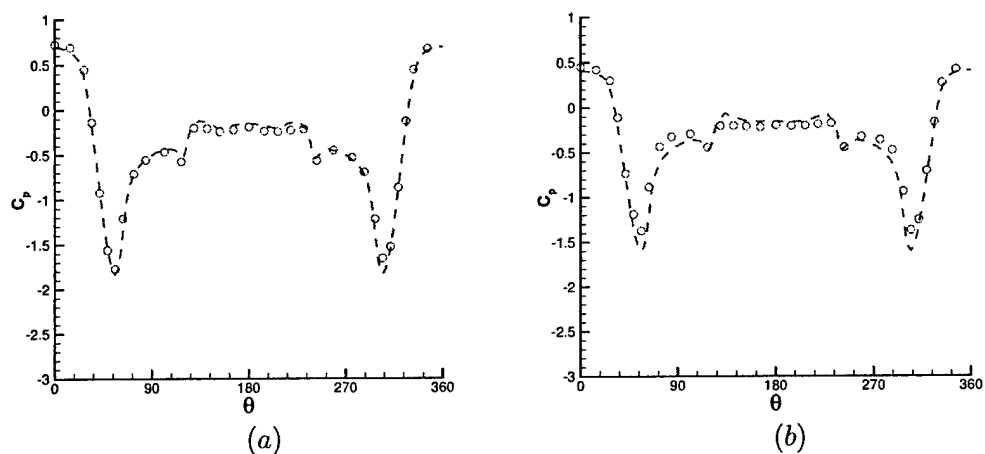
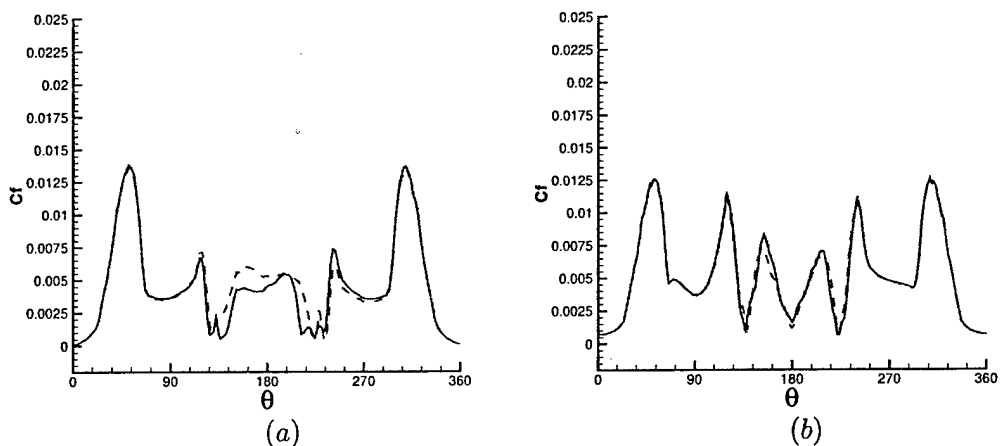Figure 26: Pressure coefficient at (a) $x/L = 0.805$ and (b) $x/L = 0.903$. o measurements; ---- URANS.
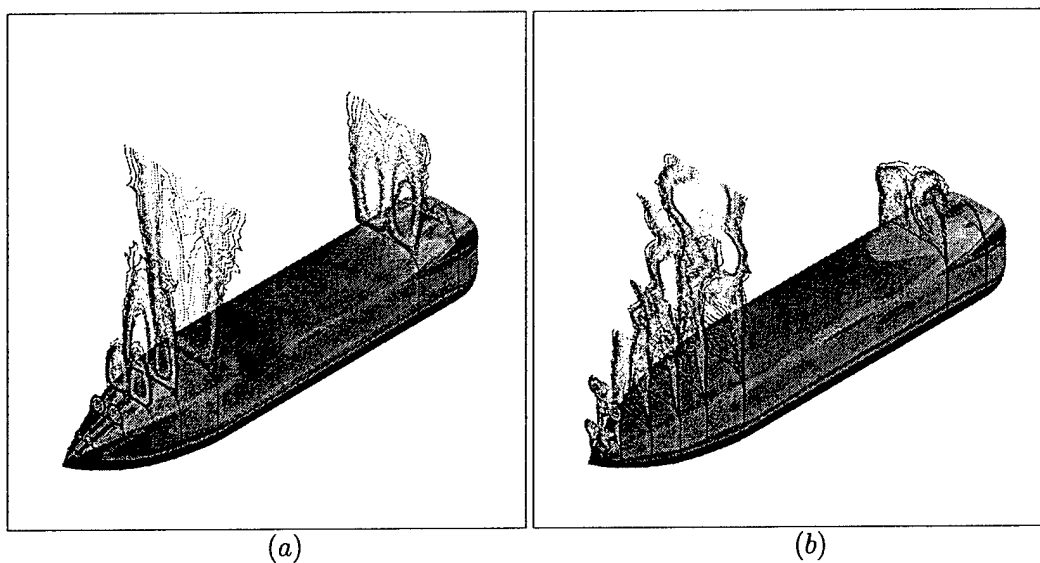
Figure 27: Ratio of the instantaneous eddy viscosity to the molecular value at the eight axial locations for which pressure measurements are available. (a) URANS; (b) DES. Geometry colored by pressure.

57
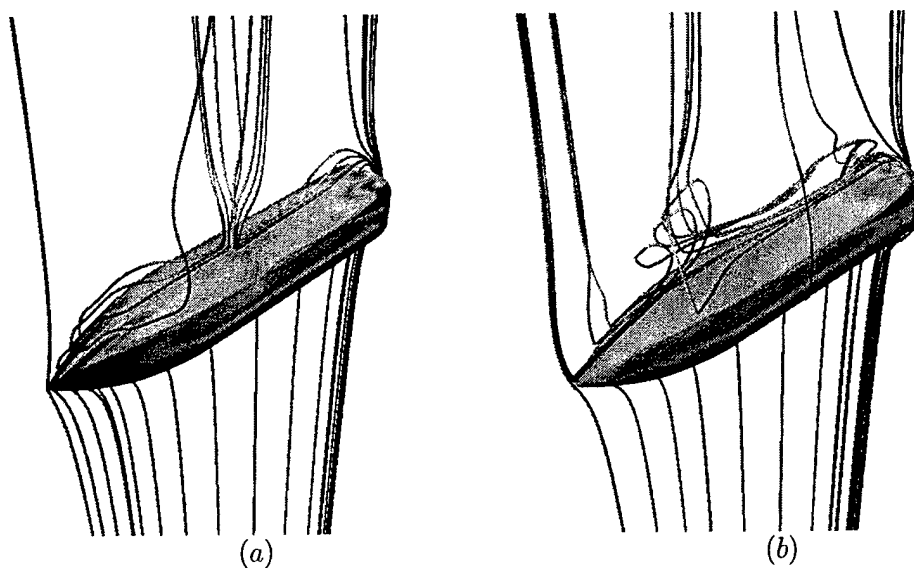
Figure 28: Streamlines around the ogive. (a) URANS (b) DES.
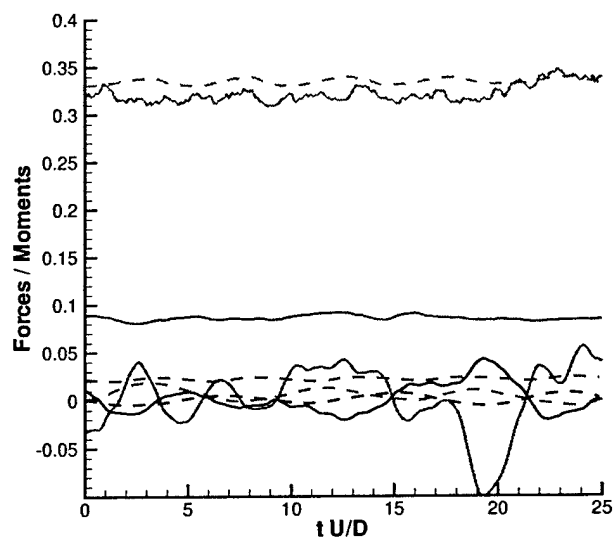
Figure 29: DES and URANS predictions of force and yawing moment histories, baseline grid. Solid lines represent DES, dashed lines represent URANS case. Green: streamwise (drag) force; red: axial force; blue: lateral (side) force; black: yawing moment.

58
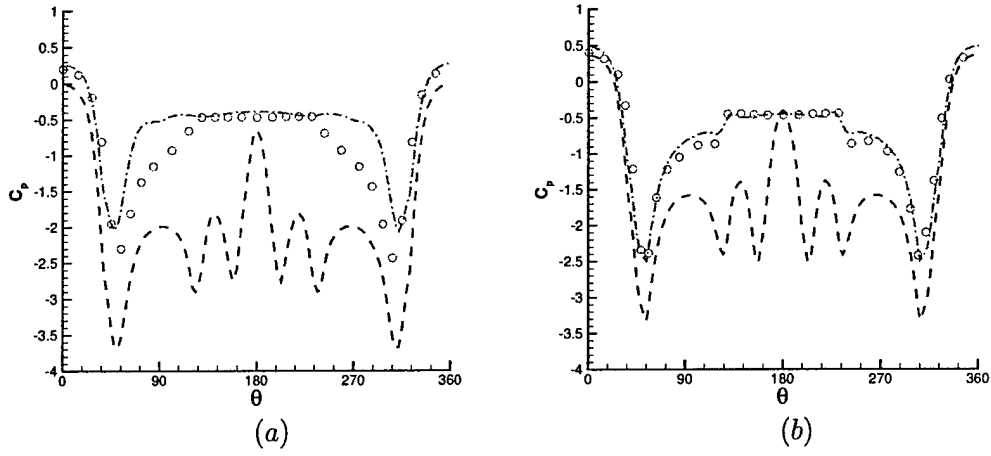
Figure 30: Pressure coefficient at (a) $x/L = 0.027$ and (b) $x/L = 0.055$.○ measurements;— · — DES; ---- URANS.
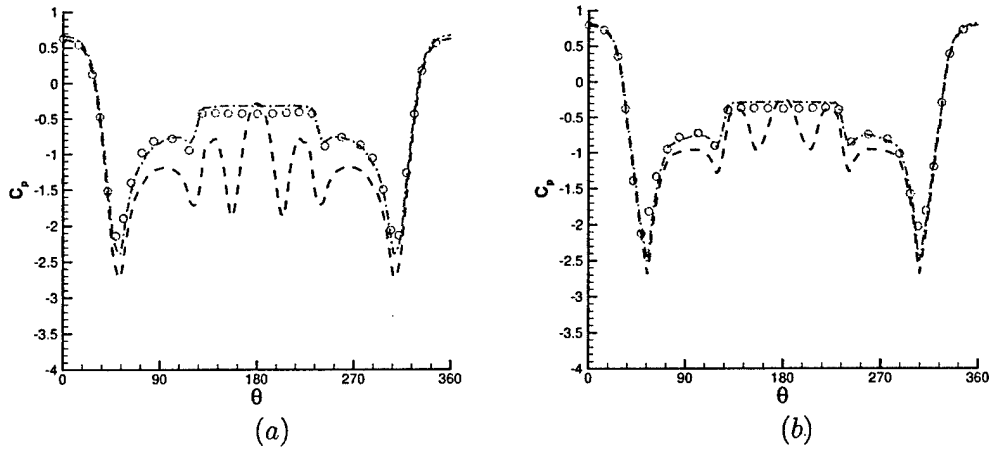
Figure 31: Pressure coefficient at (a) $x/L = 0.111$ and (b) $x/L = 0.166$. ○ measurements;— · — DES; ---- URANS.
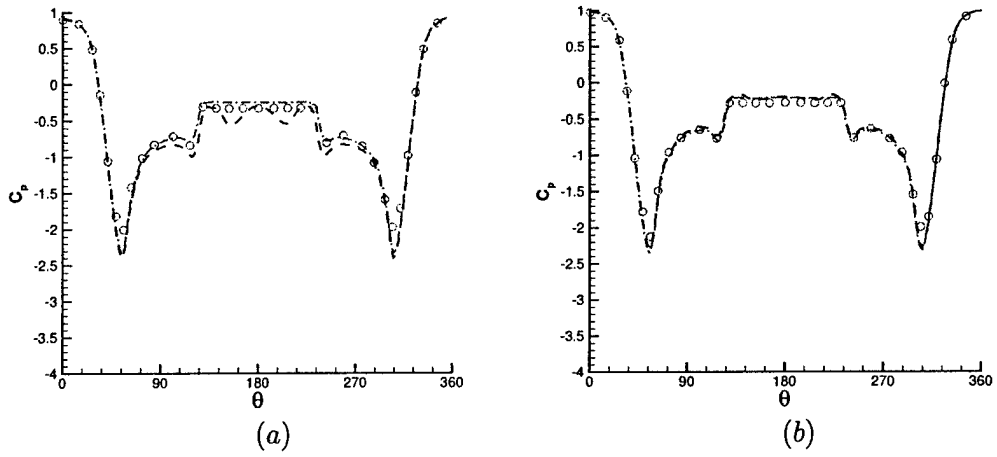
Figure 32: Pressure coefficient at (a) $x/L = 0.222$ and (b) $x/L = 0.305$. o measurements;—·— DES; ---- URANS.
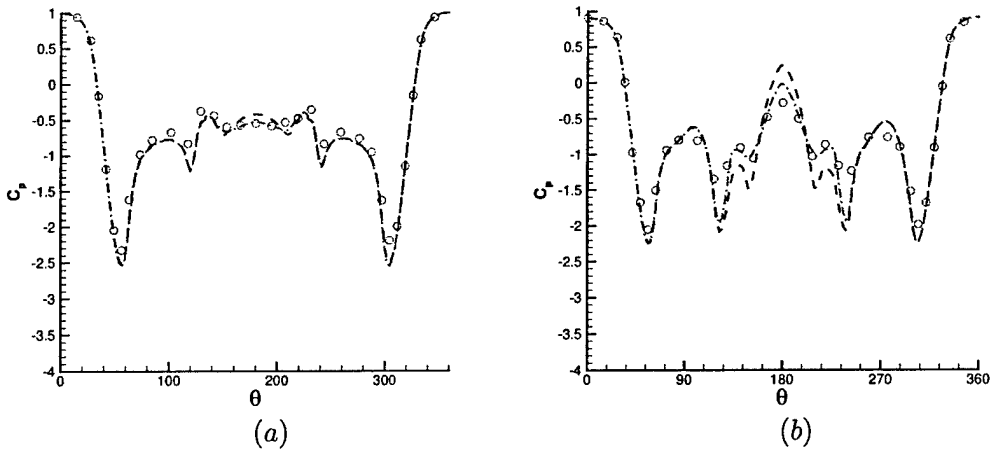
Figure 33: Pressure coefficient at (a) $x/L = 0.805$ and (b) $x/L = 0.903$. o measurements;—·— DES; ---- URANS.
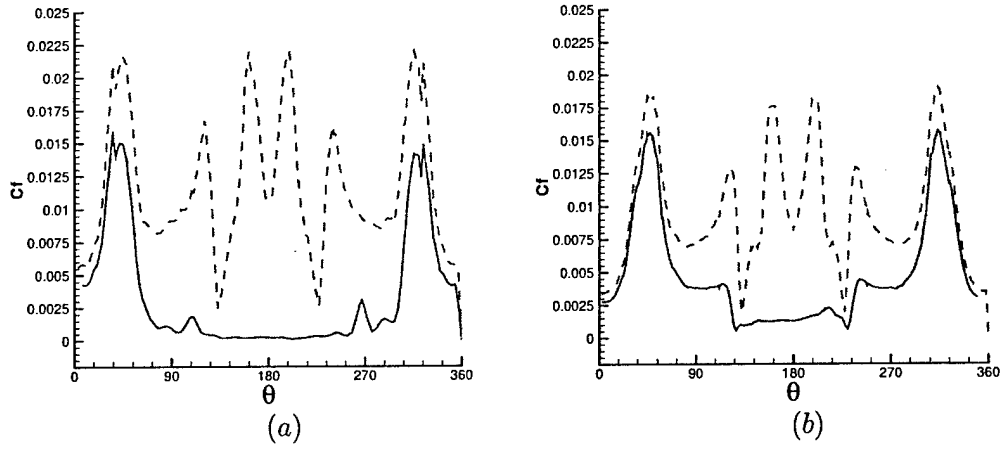
Figure 34: Skin friction coefficient at (a) $x/L = 0.027$ and (b) $x/L = 0.055$. o measurements;—— DES; ---- URANS.
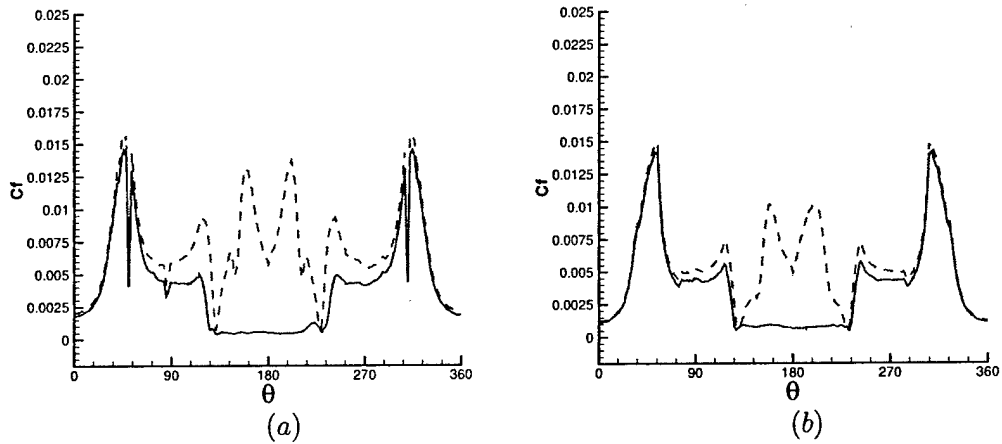
Figure 35: Skin friction coefficient at (a) $x/L = 0.111$ and (b) $x/L = 0.166$.o measurements;—— DES; ---- URANS.

Figure 36: Skin friction coefficient at (a) $x/L = 0.222$ and (b) $x/L = 0.305$. o measurements;—— DES; ---- URANS.

Figure 37: Skin friction coefficient at (a) $x/L = 0.805$ and (b) $x/L = 0.903$. o measurements;—— DES; ---- URANS.
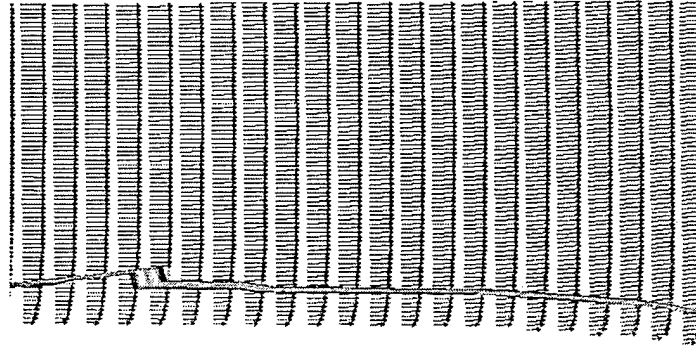
Figure 38: RANS-LES interface in the vicinity of the separation.
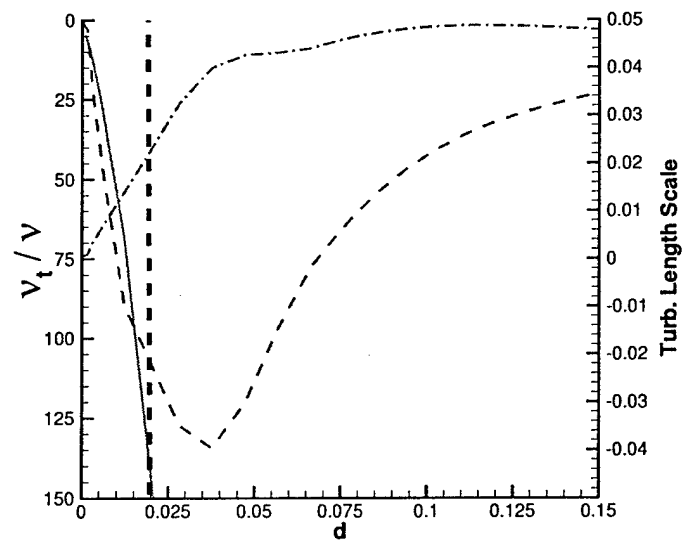
Figure 39: Eddy viscosity ratio and turbulent scale on the baseline grid. —·— turbulent length scale; ---- eddy viscosity ratio for DES; ——— Eddy viscosity ratio for RANS; ——— RANS-LES interface.
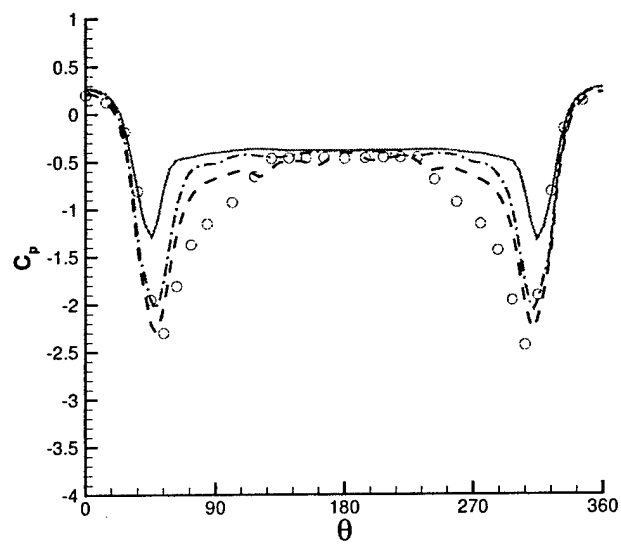
Figure 40: Effect of grid refinement on the pressure coefficient at $x/L = 0.207$. o measurements; —— coarse grid; —·— baseline grid; ---- fine grid.
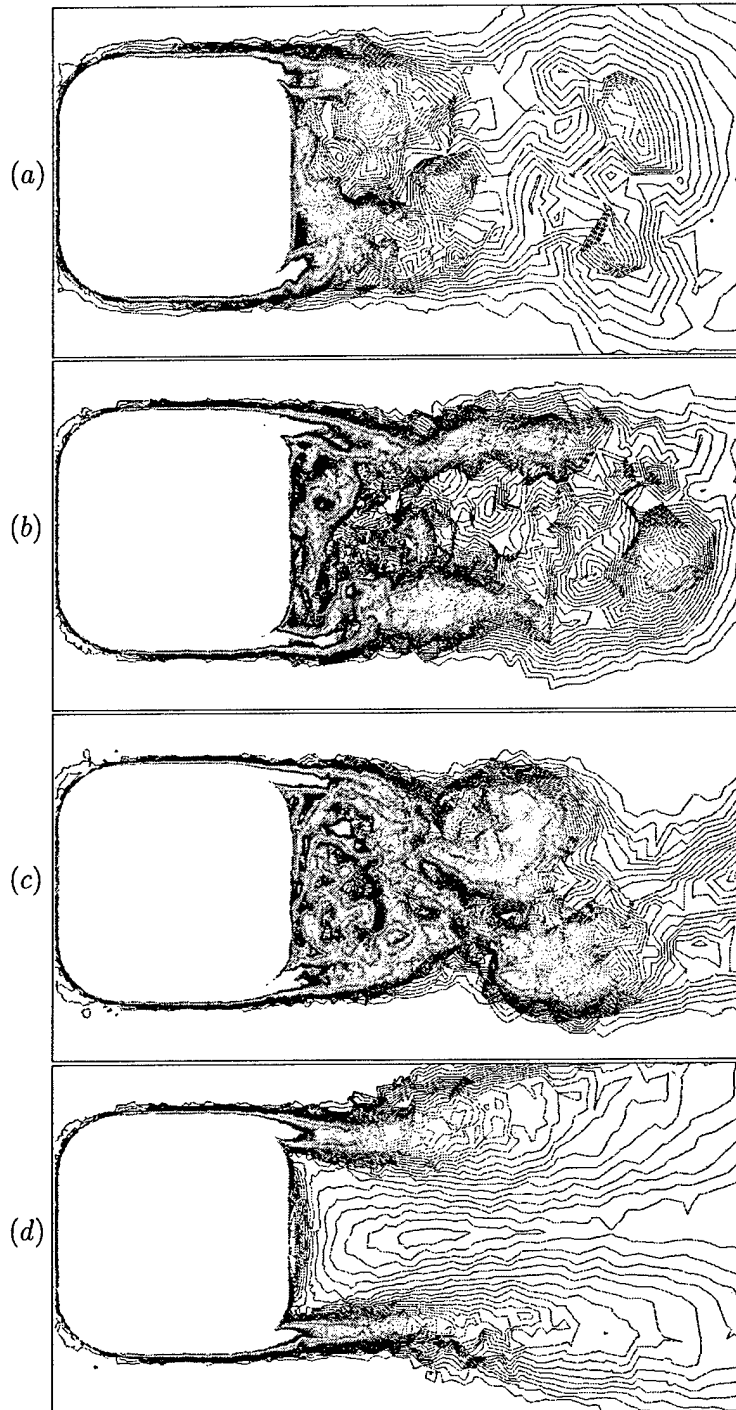
Figure 41: Contours of the instantaneous vorticity at the center of the ogive, $x/L = 0.5$. (a) coarse grid (b) baseline grid (c) fine grid (d) URANS (baseline grid).

65
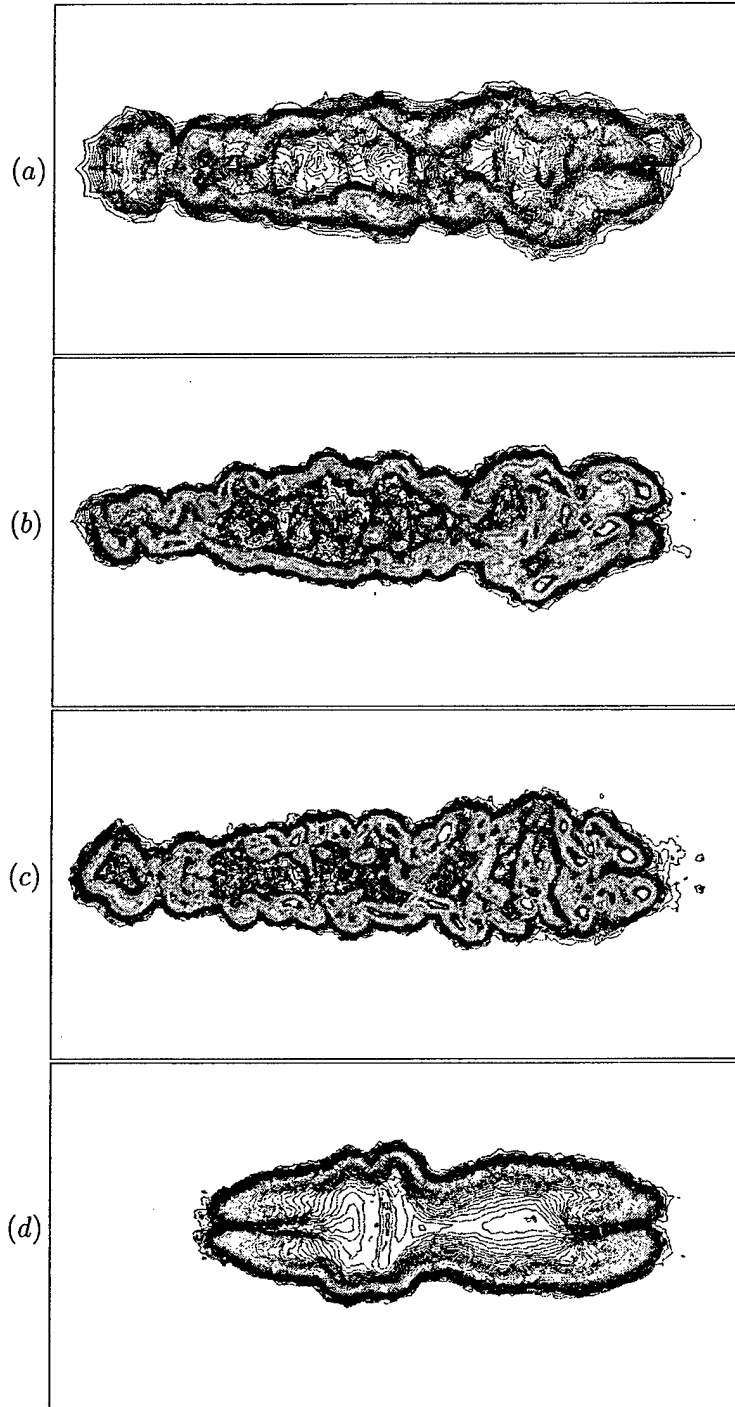
Figure 42: Contours of the instantaneous vorticity in the plane $y = D/2$, view is normal to the freestream velocity. (a) coarse grid (b) baseline grid (c) fine grid (d) URANS (baseline grid).

66
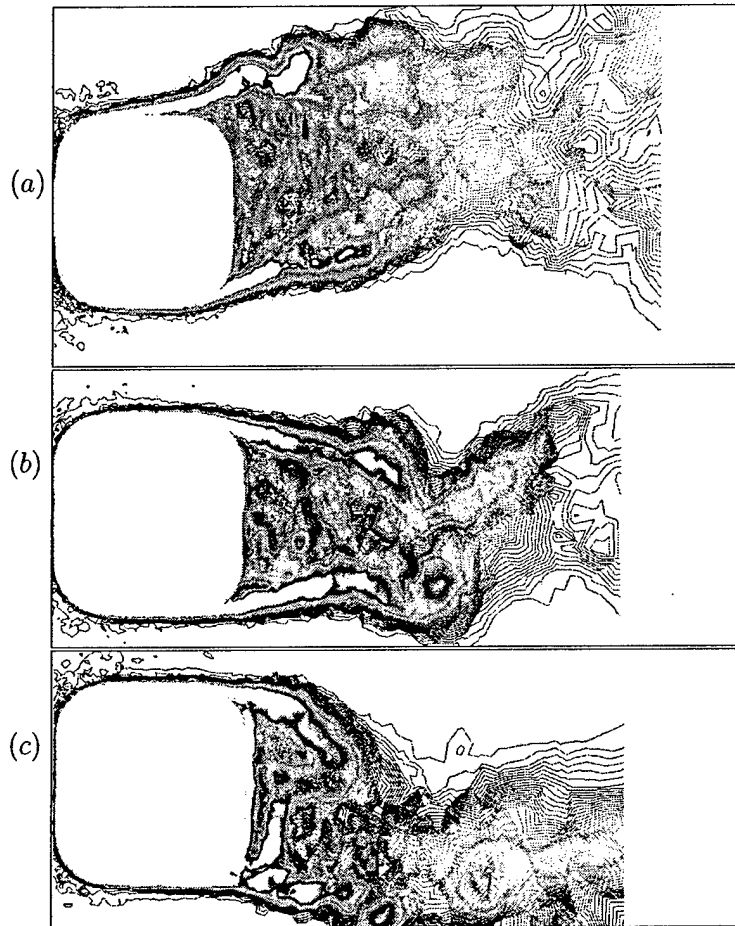
Figure 43: Contours of the instantaneous vorticity in three axial planes for DES prediction of the flow with rotary motion. (a) $x/L = 0.222$ (b) $x/L = 0.5$ (c) $x/L = 0.805$.
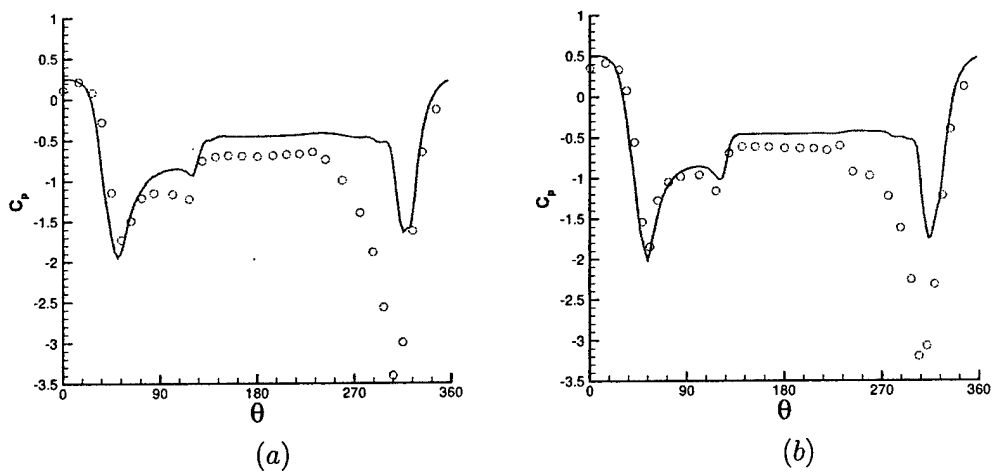
Figure 44: Pressure coefficient at (a) $x/L = 0.027$ and (b) $x/L = 0.055$, spin coefficient, $\Omega D/(2U_\infty) = 0.2$. o measurements; —·— DES; ---- URANS.
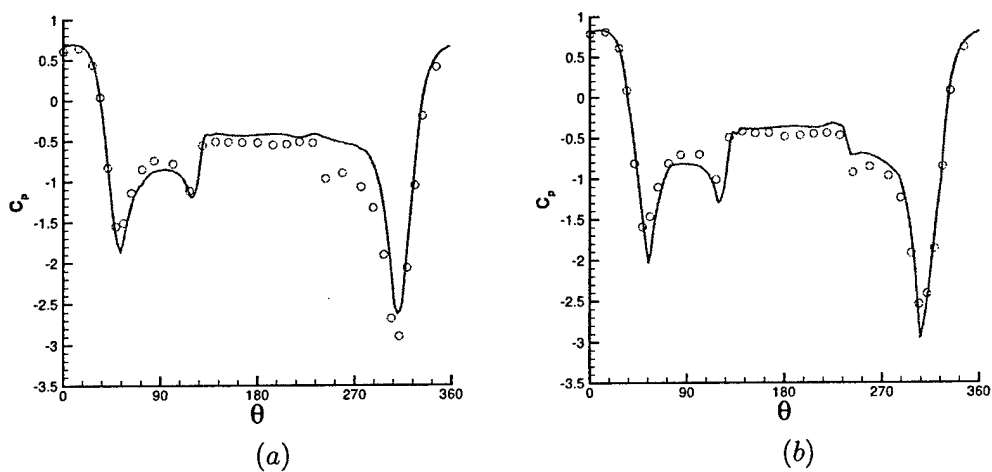

Figure 45: Pressure coefficient at (a) $x/L = 0.111$ and (b) $x/L = 0.166$, spin coefficient, $\Omega D/(2U_\infty) = 0.2$. o measurements; —·— DES; ---- URANS.
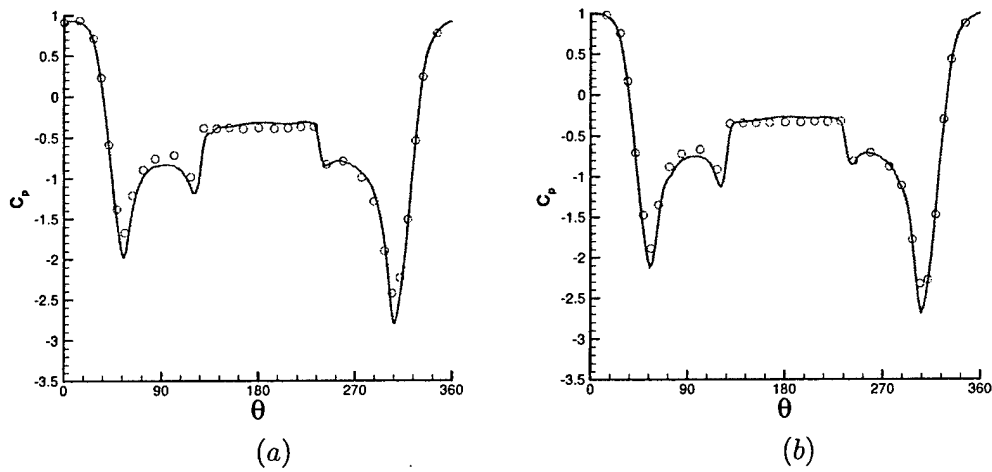
Figure 46: Pressure coefficient at (a) $x/L = 0.222$ and (b) $x/L = 0.305$, spin coefficient, $\Omega D/(2U_\infty) = 0.2$. o measurements; —·— DES; ---- URANS.
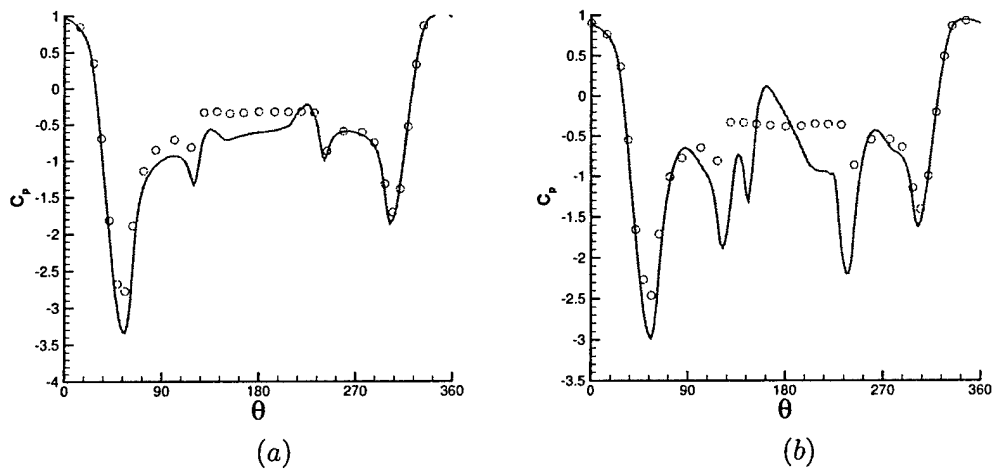

Figure 47: Pressure coefficient at (a) $x/L = 0.805$ and (b) $x/L = 0.903$, spin coefficient, $\Omega D/(2U_\infty) = 0.2$. o measurements; —·— DES; ---- URANS.